



ESCUELA POLITECNICA NACIONAL

**FACULTAD DE INGENIERIA ELECTRICA**

# TESIS DE GRADO

Tema: \*Estudio y Análisis Computacional  
de la Red de Gestión  
de Telecomunicaciones\*

ANEXOS

TESIS PREVIA A LA OBTENCION DEL TITULO DE

**Ingeniero Eléctrico en la Especialización  
de Electrónica y Telecomunicaciones**

*José Roberto Marcillo del Castillo*

Quito, Julio de 1996

A continuación tenemos la simbología de los comandos utilizados en las seis mediciones que se mencionan en el Capítulo 4, además de su forma de escritura en el computador, a fin de tener una idea clara de los tipos de información que maneja la Central AXE 10 de EMETEL, en caso de realizar alguna inspección a dicha central, y obtener valores en tiempo real de la información generada. Así tenemos:

#### Medición de Tráfico de Rutas

- TRRGE** Borra los objetos de registro de un grupo de registro  
TRRGE:TRG=trg[,R=r];
- TRRGI** Inicia un grupo de registro o añade registros a un grupo de registros  
TRRGI:TRG=trg,R=r,SI=si;
- TRRGP** Imprime los registros de un grupo de registros  
TRRGP:TRG=trg;
- TRRIC** Cambia el enrutamiento de salida de la impresión de resultados  
TRRIC:MP=mp,IO=io;
- TRRPI** Inicia un programa de medición para la medición de tráfico por rutas  
TRRPI:MP=mp,IO=io, TRG=trg;
- TRRPP** Hace una impresión de los MP's para la medición de tráfico por rutas  
TRRPP:MP=mp;

#### Medición del Tipo de Tráfico

**TRATC** Cambia el enrutamiento de la impresión de resultados de un programa de medida con medición de tipo de tráfico.

TRATC:MP=mp [,IO=io];

**TRATI** Inicia un programa de medición para el tipo de tráfico

TRATI:MP=mp [,IO=io];

**TRATP** Hace una impresión de los MPs con todos sus datos

TRATP:MP=mp;

Medición de Dispersión de Tráfico

**TRRDI** Inicia un programa de medición de la dispersión de tráfico

TRRDI:MP=mp,  $\left[ \begin{array}{l} [R=r,] \text{ TRDG=trdg} \\ R=r, \text{ TRD=trd} \\ [R=r,] \text{ TRD=ALL} \end{array} \right];$

**TRDGE** Borra los objetos de registro de los grupos de registros

TRDGE:  $\left[ \begin{array}{l} \text{TRDG=trdg [,TRD=trd]} \\ \text{RG=rg[,R=r]} \end{array} \right];$

**TRDGI** Inicia la carga de los objetos de registro en los grupos de registro

TRDGI:  $\left[ \begin{array}{l} \text{TRDG=trdg,TRD=trd} \\ \text{RG=rg,R=r} \end{array} \right];$

**TRDGP** Hace la impresión del contenido de un grupo de objetos de registro

TRDGP:  $\left[ \begin{array}{l} \text{TRDG=trdg} \\ \text{RG=rg} \end{array} \right];$

<b>TRDIC</b>	Cambia el enrutamiento de la salida de la impresión de resultados
	TRDIC:MP=mp,IO=io;
<b>TRDME</b>	Borra los criterios de selección y los objetos de registro, es decir, los objetos especificados con TRRDI
	TRDME:MP=mp;
<b>TRDMI</b>	Inicia el MP para la medición de dispersión de tráfico
	TRDMI:MP=mp [,IO=io] $\left  \begin{array}{l} \text{TRD} \\ \text{,R1} \\ \text{R2} \end{array} \right  ;$
<b>TRDMP</b>	Hace la impresión perteneciente a los MPs
	TRDMP:MP= $\left  \begin{array}{l} \text{mp,DATA} \\ \text{mp} \\ \text{all} \end{array} \right  ;$
<b>TRDRI</b>	Especifica los criterios de selección y los objetos de registro, para un programa de medición
	TRDRI:MP=mp, $\left  \begin{array}{l} [\text{TRD=trd,}] \text{RG=rg} \\ \text{TRD=trd,R=r} \\ [\text{TRD=trd,}] \text{R=ALL} \end{array} \right  ;$

#### Medición de Carga del Procesador

<b>PLSMI</b>	Inicia un programa de medición de la carga del procesador
	PLSMI:MP=mp [,IO=io];
<b>PLSMP</b>	Hace una impresión de los MPs con todos sus datos
	PLSMP:MP=mp;

#### Medición de la Calidad de Servicio

<b>SQMPI</b>	Inicia al MP para la medición de la calidad de
--------------	--

servicio además especifica los criterios de selección.

```
SQMPI:MP=mp<, IO=io><QTA=qta>
                                |INT
                                |ORG
                                |IEX
                                |,ADDR=addr
                                |R1=r1
                                |R2=r3
                                |;
```

**SQMPP** Hace la impresión de los MPs definidos para la medición de calidad de servicio.

```
SQMPP:MP=
                                |mp
                                |ALL
                                |;
```

**SQREP** Solicita una impresión de resultado antes que termine la medición.

```
SQREP:MP=mp;
```

**SQTLE** Pone en cero, los valores de BANSWER, CTIME y PDD.

**SQTLI** Especifica los valores de BANSWER, CTIME y PDD.

```
SQTLI:MP=
                                |PDD=pdd
                                |BANSWER=banswer
                                |CTIME=ctime
                                |;
```

**SQTLP** Hace la impresión de los valores de BANSWER, CTIME y PDD.

```
SQTLP;
```

**SQENE** Especifica direcciones numéricas que habían estado excluidas de la medición, pero que ya no se va a aplicar la exclusión.

SQENE:NRL=nrl,NRD=nrd;

**SQENI** Especifica las direcciones numéricas que van a ser excluidas de la medición.

SQENI:NRL=nrl,NRD=nrd;

**SQENP** Hace la impresión de las direcciones numéricas que están excluidas de la medición.

Medición de Observación de Tráfico

**TOBSI** Inicia la observación de tráfico. Solo se acepta si antes se ha enviado el comando TOCOS. El parámetro NOFILE indica que no habrá salida hacia un archivo (file).

TOBSI:[NOFILE];

**TOBSP** Hace una impresión de los criterios especificados en TOCOS.

TOBSP;

**TOCOR** Borra los criterios de selección especificados con TOCOS.

TOCOR;

**TOCOS** Define los criterios de selección de la observación.

TOCOS:                   |ORG           |           |TE  
                          |IEX           |           |OEX  
                          |R1=r1       |           |R2=R2  
                          |           |           |TRD  
                          |           |           |           |,QTA=qta;  
                          |TE           |  
                          |OEX       |

R2=r2
TRD=trd

**Comandos aplicables a cualquier medición**

**TRIDP** Hace la impresión de todos los MPs disponibles para su utilización.

TRDIP:[ALL];

**TRTSE** Borra el cronograma, y si se indica MPE en el comando se borra el MP completo.

TRSTE:MP=mp[,mpe];

**TRTSI** Define un cronograma

TRTSI:MP=mp[,NRP=nrp][,RPL=rpl][DATE=date][,TIME=time]  
[,NDAYS=ndays][,DCAT=dcap];

**TRTSP** Hace una impresión de los cronogramas de los MPs

TRTSP:MP=mp;

**IOFAT** Sirve para extraer la información de los archivos del disco duro del IOG11.

IOFAT:FILE=file [,NF];

**IOTIP** Sirve para extraer información de los archivos de las centrales telefónicas del IOG3.

IOTIP:IO=CT\_xc,PRINT=ALL,NF;

**INTRODUCCION.**- En el momento actual, el entorno Windows se constituye como una de las tendencias dentro del mundo informático, debido a su versatilidad, la amigabilidad que ofrece a todo tipo de usuarios de software, las posibilidades que ofrece de compartir datos entre distintas aplicaciones y la gran cantidad de controles que ayudan al desarrollo de poderosas aplicaciones en diferentes versiones que pueden correr bajo este entorno.

En cierto modo, lo que es Windows depende de la característica del usuario, esto es, sea como programador o como usuario común, ya que Windows puede ser un interfaz global que interactúa con la ejecución de aplicaciones o un sistema operativo multitarea de orientación gráfica, constituido por una colección de varios centenares. Para el programador, Windows es una enorme caja de herramientas de servicios interrelacionados que; cuando se usan en forma correcta, permite la creación de programas de aplicación que comparten un interfaz común.

El objetivo de Windows es el permitir que una persona, familiarizada de forma básica con el sistema, pueda usarlo para ejecutar prácticamente cualquier aplicación, sin ninguna formación previa. Teóricamente, si un usuario puede ejecutar un programa para Windows, entonces puede ejecutar cualquiera. Obviamente que, en realidad, la mayoría de los programas que prestan cierta utilidad, siempre requerirán que el usuario adquiera algún tipo de formación para poder utilizarlos en forma efectiva.



Windows es un sistema orientado a gráficos, lo cual significa que proporciona un interfaz gráfico de usuario (GUI). Aún cuando, el hardware de gráficos y los modos de video son muy diversos, Windows se hace cargo de muchas de estas diferencias.

En muchos sistemas operativos, cuando se escribe un programa, es este mismo, el que debe iniciar la interacción con el sistema operativo. Por ejemplo, en un programa para DOS, es el programa quien solicita las funciones de entrada/salida, es decir, el sistema operativo no llama al programa; sin embargo, en gran parte, Windows trabaja en contrasentido, esto es, windows es quien llama al programa.

Es de esta filosofía, donde nacen los dos tipos de programación para la creación de aplicaciones bajo un entorno Windows, así:

**\* Programación orientada a objetos:** Es la que considera que cada ventana es un objeto , y que todas las interacciones con este objeto se efectuarán a través de funciones miembro especiales, que responden a los mensajes enviados por Windows, bajo esta premisa, cada nueva ventana tiene que ser definida por el usuario a través de herencia de objetos preestablecidos y de la programación de funciones miembro de estas clases.

**\* Programación orientada a eventos:** En esta programación no se pueden definir nuevas clases a partir de otras ya preestablecidas, por el contrario, una ventana es un objeto ya definido, y el programador debe codificar los eventos que podrían darse en un

tiempo determinado, ya sea por eventos realizados a través del sistema o ejecutados por el usuario.

**VISUAL BASIC.**- Es un sistema gráfico de desarrollo de aplicaciones para entorno Windows. En este lenguaje, los programas son creados de forma totalmente visual. Con la ayuda del ratón, el programador extrae de un cuadro de utilidades, los objetos tales como: los botones, barras de desplazamiento y otros controles posicionándolos directamente en la pantalla.

Después de la preparación de las pantallas o ventana de un programa, de forma directa e inmediata, sin escribir ni una sola línea de código, el programador describe las funciones de cada objeto y su relación con los estímulos externos, como pulsaciones de teclas o botones del ratón.

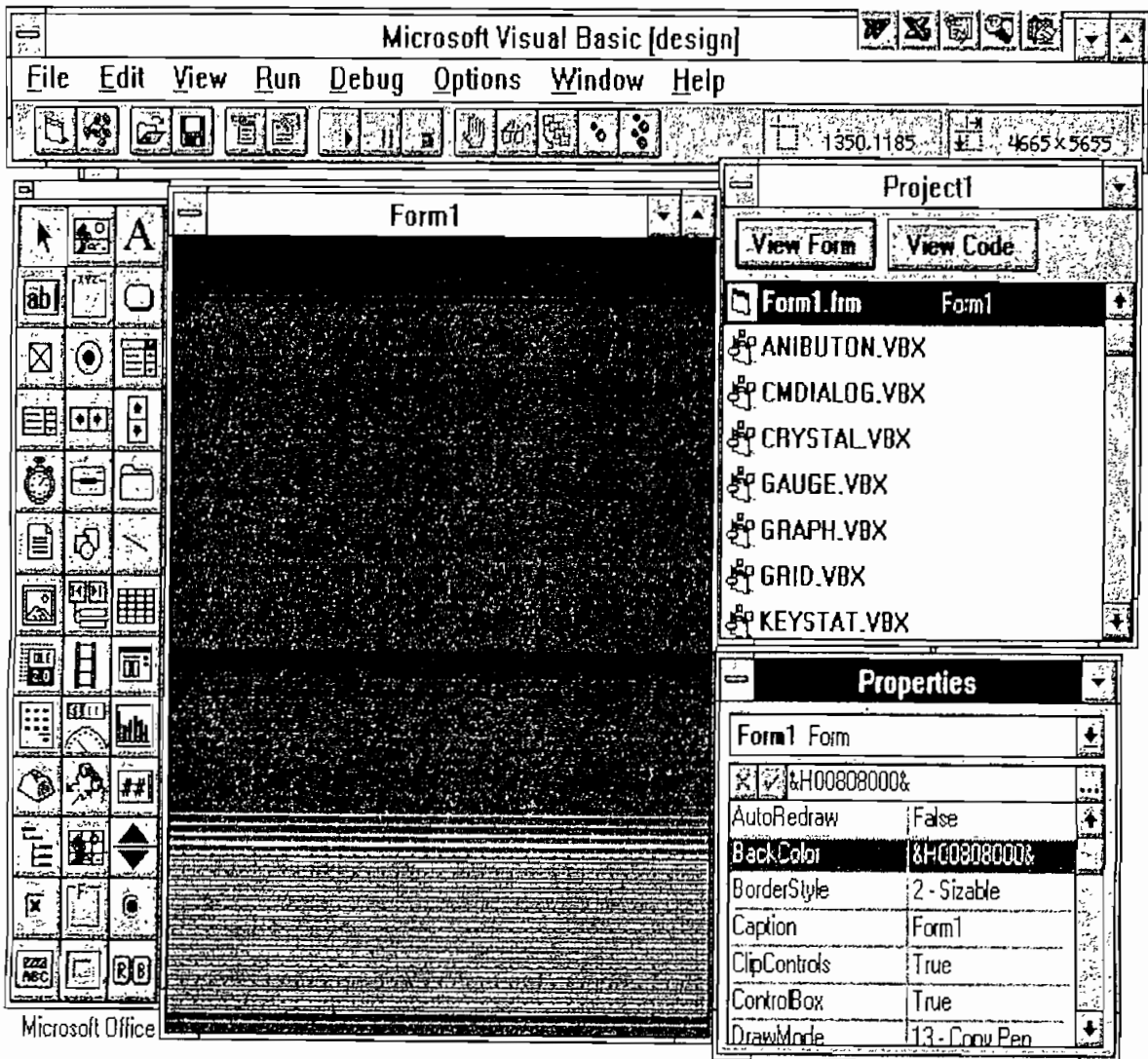
Con este sistema, puede desarrollarse cualquier tipo de aplicación basada en el entorno Windows, incluyendo sistemas administrativos, herramientas, utilidades, aplicaciones front-end para procesos de los datos de un main-frame, servidores o procesadores locales, y hasta productos de software comercial, ya que se pueden crear y compilar programas ejecutables.

\* **Ambitos de programación.**- En el momento en que ingresamos a Visual Basic, se nos presenta una pantalla como la observada en la figura B.1.

- En su primera línea se encuentra el título de la pantalla: Microsoft Visual Basic (**design**); esta última palabra significa que nos encontramos en modo **Proyecto**. Cuando se ejecute algún

programa, la palabra entre paréntesis pasará a ser run.

- En la línea siguiente, se encuentra la barra de menú (Menu Bar), con todas las opciones que podemos utilizar.



- En la tercera línea se encuentra la barra de herramientas (toolbar), esta provee un acceso rápido a los comandos más utilizados en ambiente de diseño; solamente hay que hacer click sobre el icono, para ejecutar la acción.
- En la primera columna, se ubica el cuadro de utilidades (toolbox), que provee un conjunto de herramientas para usar los controles sobre una forma (form).
- Luego tenemos la ventana (form minimizada), esta sirve como una window para actuar como interface de la aplicación. Aquí se pueden agregar controles, gráficos y cuadros que se utilizarán cuando el programa se ejecute.
- La otra ventana es la de propiedades (Properties Window), la cual lista las propiedades para el control o form seleccionado. Una propiedad es el valor de un objeto, tal como su tamaño, su título (caption), o color.
- La ventana de proyectos (Project Window), ésta lista los forms, módulos de código y archivos de control establecidos en el proyecto actual. Un proyecto es una colección de archivos, que se utilizan para generar la aplicación.

\* **Cómo crear una aplicación?** El primer paso diseñar el medio de comunicación (interface) entre el usuario y la aplicación. Así tenemos:

- Crear la forma;
- Añadir los controles a la forma.

El siguiente paso consiste en especificar las propiedades para las formas y para los controles. Por ejemplo, títulos

(caption) deseados para los objetos, sus nombres (names), valores iniciales, colores, etc.

\* **Otras facilidades de Visual Basic.**— Este lenguaje en ambiente de trabajo, provee varias herramientas para facilitar el diseño de cualquier aplicación gráfica.

- El conjunto de todos los archivos (módulos, formas, etc.) que forman una aplicación se denomina **Proyecto (Project)**.

- Para visualizar una forma de un proyecto, primero se selecciona haciendo click sobre el nombre de la misma, en la ventana correspondiente al proyecto, y después se hace clic sobre el botón **View Form** .

- Para ver el código de un módulo o de una forma, se selecciona en la ventana correspondiente a la aplicación, y después se hace clic sobre el botón **View Code**.

- Una **forma (form)** es una ventana que sirve de fondo para los controles y los gráficos situados sobre la misma. Se pueden utilizar tantas formas como se necesiten y dependiendo de la utilidad que se las puede dar, estas serán de diferentes tipos. Así, se puede crear una forma que contenga un gráfico, para visualizar información, o para aceptar datos.

- Los controles son objetos que dibujamos sobre una forma con fines de aceptar o visualizar datos. Para dibujar los controles en una forma, Visual Basic pone a nuestra disposición un panel de utilidades (**ToolBox**). Con él, se pueden dibujar controles como: botones, cajas de texto y etiquetas. Cada objeto lleva asociado un conjunto de propiedades (Nombre, Color, etc.). Para

la especificación de las mismas, utilizamos la ventana **Propiedades (Properties)**.

- Cualquier aplicación que se cree en Visual Basic es conducida por sucesos/eventos, esto significa que las formas y los controles pueden responder a sucesos, tales como, un clic de un ratón o un tiempo transcurrido. Para hacer que la aplicación responda a sucesos tenemos que unir código a las formas y a los controles. El código que se une a un objeto se denomina **procedimiento conducido por un suceso** o simplemente **procedimiento**.

Para unir un procedimiento a un objeto hay que escribir el código correspondiente a dicho procedimiento en la ventana de código asociada con el objeto. Para visualizar la ventana código de una determinada forma, hay que hacer un doble clic dentro del área correspondiente a esa forma, y haciendo doble clic sobre esta área correspondiente a un control, se visualiza la ventana de código asociada a ese control.

Un procedimiento está formado por un nombre del objeto, un nombre de suceso y una o más secuencias.

Todo código asociado con una forma y con sus controles, se almacena con esa forma. Cuando una aplicación consta de varias formas y se quiere compartir código entre ellas, hay que tener el código en un **Módulo**. Un módulo es un archivo que contiene los procedimientos que son compartidos por las formas de una aplicación.

\* **Un lenguaje de Alto nivel.**- Incluye un lenguaje fácil de utilizar que proporciona muchas de las características de MS QBasic o MS QuickBasic. Algunas de ellas son:

**- Bloques if...then...else**

```
If Var1>0 and Var2>0 then
    Var3=Var2/Var1
Else
    Var3=0
Endif
```

**- Bucles**

```
Do While i<=100
    Print i
    i=i+1
```

Loop

**- Ocho tipos de datos: Integer, Single, Double, Currency, String, Form, Control.****- Funciones Matemáticas y de Cadena**

```
Abs(Var1-Var2)
Print UCase(Libro$)
```

Como ayuda para revisar el código rápidamente, Visual Basic proporciona:

- Verificación automática de la sintaxis;
- Utilidades para depurar código;
- Ventana inmediata que permite probar una línea de código o verificar una variable.

\* **Menús.**- Cuando en una aplicación se utilizan muchas órdenes, VB permite agruparlas en un menú; esta se diseña utilizando la

ventana de diseño de menús, para visualizar esta ventana hay que ejecutar la orden **Menú Design** del menú **Window**.

\* **Utilización de la Ayuda (Help).**— Visual Basic tiene instalada una ayuda en línea que da información acerca de las propiedades, sucesos, mensajes de error, sentencias, teclado y entorno de VB. Para obtener la ayuda se puede utilizar el índice proporcionado por el menú **Help** de la barra de menús de VB, o se puede seleccionar el elemento deseado en el entorno de VB y pulsar **F1**.

Cuando se está dentro de la ayuda, tenemos lo siguiente:

- Se utiliza la orden **Search** para buscar un término o una palabra clave del lenguaje;
- Haga clic sobre una frase o una palabra subrayada para ver la ayuda referente a la misma, o;
- Haga clic sobre una palabra subrayada con trazos discontinuos para ver su definición.

Cuando se visualice un mensaje y se desee ayuda sobre el mismo, se pulsa **F1**.

Cuando se esté dentro de la ventana de código, se selecciona una palabra del lenguaje, y se pulsa **F1** para obtener ayuda sobre la misma.

Para obtener ayuda sobre un menú o una herramienta de Visual Basic, se selecciona y se pulsa **F1**.

\* **Controles en Visual Basic.**— Los controles son objetos gráficos insertados en una pantalla o ventana (**form**). Estos son utilizados



básicamente como interfaz entre las entradas o salidas de los usuarios de las aplicaciones.

Cada uno de los controles posee su propio conjunto de propiedades o atributos, que definirán su apariencia final en la pantalla o ventana, así tenemos: posición, colores, o tamaño. Además, define su funcionalidad, por ejemplo, cómo responderá a las entradas de los usuarios o cómo se emitirán los mensajes en la pantalla?.

Responden a sucesos, para los cuales puede ser necesario escribir algún código. Estos sucesos pueden ser activados tanto por el usuario como por el sistema.

Palabras clave en VB pueden ser utilizadas para mostrar los controles, diseñar gráficos y textos. Estas palabras clave son denominadas métodos. Los controles son creados mediante el cuadro de utilidades (toolbox)

A continuación se lista una serie de controles con sus principales características:

- **Cuadro de Verificación (Check Box).**- Representa una operación que puede ser activada o desactivada. Esta opción sirve para que el usuario indique si alguna cosa es verdadera o falsa. Este control puede ser utilizado también para mostrar opciones múltiples, en donde el usuario deberá seleccionar más de una.
- **Cuadro Combinado (Combo Box).**- Cuadro de textos con una lista de opciones; permite a los usuarios escribir textos en un cuadro de textos, o seleccionar un ítem de una lista tipo descendente-oculta (drop-down).
- **Botón de Mandato (Command Button).**- Control que el usuario

puede pulsar para ejecutar una operación determinada o seleccionar una opción de una aplicación. Cuando se selecciona aparece en la pantalla como si se hubiera presionado el botón realmente.

- **Cuadro de lista de directorios (Directory List Box).**- Control que muestra la estructura de los directorios del disco seleccionado. Para que el usuario tenga acceso a los ítems de la lista mostrada, se deben ajustar las propiedades List, ListCount y ListIndex a True (verdadero).

- **Cuadro de lista de unidades (Drive List Box).**- Control que muestra y permite cambiar entre los discos disponibles en el equipo. Para que el usuario tenga acceso a los ítems de la lista mostrada, se deben ajustar las propiedades List, ListCount y ListIndex a True.

- **Cuadro de lista de archivos (File List Box).**- Muestra una lista de todos los archivos disponibles en el directorio seleccionado. Para que el usuario tenga acceso a los ítems de la lista mostrada, se deben ajustar las propiedades List, ListCount y ListIndex a True.

- **Marco (Frame).**- Proporciona un medio gráfico y funcional para los controles. Se puede utilizar para subdividir la pantalla gráficamente, por ejemplo, para separar un grupo de botones de opción de otro de opción o controles.

- **Barras de desplazamiento horizontal y vertical.**- Estos dos controles son utilidades gráficas, que permiten al usuario navegar a través de una lista de ítems o de una gran cantidad de información, e indicar la posición actual en esta lista.

Una barra de desplazamiento, también puede ser usada como

indicador de velocidad o cantidad, por ejemplo, la visualización de la cantidad de CPU que se está utilizando en ese momento. Cuando estas barras son utilizadas como indicadores de cantidad, las propiedades **Max** y **Min** deben ser ajustadas para mostrar el rango apropiado para el control.

Para especificar una cantidad de alteraciones a mostrar en una barra, se debe utilizar la propiedad **LargeChange** para el click en la barra, y la propiedad **SmallChange** para el click en las flechas de los extremos de la barra. La propiedad **Value** es incrementada o decrementada por la cantidad ajustada en las propiedades **LargeChange** y **SmallChange**. Se puede posicionar un cuadro de desplazamiento durante la ejecución, ajustando la propiedad **Value** a un valor entre 0 y 32767.

- **Etiqueta (Label).**- Identifica un elemento de un cuadro de diálogo. Puede ser utilizado para mostrar textos extensos, que no necesitan modificaciones durante la ejecución de la aplicación. Se puede escribir códigos para alterar la etiqueta, como respuesta a sucesos durante la ejecución de la aplicación. Por ejemplo, si una aplicación se demora algunos instantes para efectuar un proceso, puede que se desee crear un mensaje para mostrar el estado de este.

También puede ser usada para identificar un control, como cuadro de texto (**TextBox**), que no posea su propiedad **Caption**.

- **Cuadro de Lista (List Box).**- Muestra una lista de opciones de la cual se podrá seleccionar una. Si el número de ítems excede al que puede ser mostrado, se insertará una barra de despla-

miento automáticamente, haciendo posible que el usuario pueda desplazar la lista hacia arriba o hacia abajo, para visualizar todos sus ítems.

- **Menú (Menu).**- Este control muestra la barra de menús que se proyecta para la aplicación. Los ítems de un menú pueden incluir mandatos que el usuario puede seleccionar para ejecutar una acción, nombres de submenús y barras espaciadoras. Se puede crear hasta seis niveles de menú.

Se ajusta la propiedad **Caption** para mostrar el texto. Para crear una barra espaciadora en el menú se describe un guión en el cuadro **Caption** en la ventana de proyectos de menús (**Menu Design**).

- **Botón de Opción (Option Button).**- Muestra una opción que puede ser activada o desactivada. Si se encuentra junto con otros botones de opción formando un grupo, el usuario solamente podrá seleccionar una de las propiedades mostradas. Se puede agrupar los botones de opciones en un **frame**, en un cuadro de imagen (**picture box**), o directamente en un **form**.

Los botones de las opciones pueden parecer similares a los cuadros de verificación, pero poseen una diferencia importante: cuando el usuario selecciona un botón de opción, los otros botones del mismo grupo son cancelados automáticamente. En cambio, se puede seleccionar tantos cuadros de verificación (**check box**) como se desee.

- **Cuadro de Imagen (Picture Box).**- Puede mostrar gráficos a partir de archivos **Bitmap** (mapa de bits), **metafiles** o también **iconos**. Se debe notar que sólo se mostrará lo que cabe en el

rectángulo que se proyectó para el cuadro de imagen.

Para que el tamaño del Picture Box sea automáticamente adecuado a todo lo que debe ser mostrado en su interior, se debe ajustar la propiedad **AutoSize** a **True**.

- **Cuadro de Texto (Text Box).**- También llamado campo de edición, puede mostrar las informaciones que especifica el proyecto, así como las informaciones escritas por los usuarios.

Para mostrar varias líneas de texto durante la ejecución, se debe ajustar la propiedad **MultiLine** a **True**.

Para mostrar entradas numéricas, se debe utilizar la función **Val**.

También puede ser ajustado como un cliente de un enlace de una conversación **DDE de Windows**. Para esto se debe ajustar la propiedad **LinkTopic** para establecer el enlace, la propiedad **LinkItem** para especificar el ítem de conversación y la propiedad **LinkMode** para activar el enlace.

- **Temporizador (Timer).**- Es un control que se puede programar para ejecutar los códigos a intervalos regulares de tiempo, funciona en forma independiente al usuario. También cabe hacer notar que este control es invisible en la ejecución del programa.

\* **Tipos de datos y operadores en Visual Basic.**- A continuación tenemos los principales tipos de datos soportados; además de los operadores con los cuales se puede trabajar.

En VB el tipo de dato por defecto es **Variant**, un tipo de dato bastante flexible.

Un operador puede ser considerado como una conjunción lógica

utilizada para evaluar dos o más expresiones.

TIPO	SIGNO	TAMAÑO	RANGO
ENTERO	%	2 bytes	-32768 a 32767
ENTERO LARGO	&	4 bytes	-2,147E9 a 2,147E9
SIMPLE PREC.	!	4 bytes	-3,402E38 a -1,401E-45 1,401E-45 a 3,402E38
DOBLE PREC.	#	8 bytes	-1,7979E308 a -4,94E-324 4,9E-324 a 1,7977E308
NORMAL	@	8 bytes	-9,2233E14 a 9,2233E14
STRING	\$	1 byte/ caracter	0 a 65500 aprox.
VARIANT		Date/Time, punto flo- tante, nu- mérico o string	0 a 65500 aprox.

Se pueden declarar arreglos con cualquiera de estos tipos.

Dentro de los operadores tenemos:

- **And:** El resultado es verdadero (-1), si ambas expresiones son verdaderas; caso contrario el resultado será falso (0).
- **Not:** El resultado es el inverso de la expresión, niega la misma.
- **Or:** Este operador entregará verdadero, si ambas o al menos una de las expresiones es verdadera, de lo contrario es falsa.
- **Xor:** El valor del resultado será falso, si ambas expresiones son verdaderas o falsas; caso contrario será verdadero.

\* **Palabras reservadas en Visual Basic.**— Las palabras clave son palabras reservadas para el lenguaje, es decir, no pueden ser utilizadas con ningún otro propósito. Sirven para complementar

declaraciones, o poseen significado propio si son consideradas aisladamente, determinando la tarea que debe ser ejecutada.

A continuación tenemos estas palabras con las declaraciones que pueden acompañarlas, así:

- As** - Declare, Dim, Global, Name, Open, ReDim, Static.
- ByVal** - Call, Declare, Function, Sub.
- If - Else**
- Shared** - Dim, Open, ReDim.
- Step** - Circle, For...Next, Line, Pset.
- To** - Dim, For...Next, Global, Lock, ReDim, SelectCase, Static, Unlock.

\* **Declaraciones en Visual Basic.**- Las declaraciones son los códigos que asignan nombres a constantes o variables, y definen sus atributos. Las declaraciones se escriben para los forms o los módulos. Para escribir estas declaraciones, se debe abrir la sección de declaraciones del proyecto a través de la ventana de códigos, y seleccionar General.

- AppActivate** - Instrucción que activa una ventana de la aplicación.
- Beep** - Instrucción que hace sonar la alarma del PC.
- Call** - Instrucción que transfiere el control del programa a un procedimiento en una DLL (Dynamic Link Library) o a un procedimiento Sub en VB.
- ChDir** - Instrucción que cambia el directorio actual al especificado.
- ChDrive** - Instrucción que cambia del disco actual al especificado.

---

<b>Close</b>	- Instrucción que concluye una transacción de entrada/salida con un archivo.
<b>Const</b>	- Instrucción que permite la utilización de constantes simbólicas en lugar de valores.
<b>Date\$</b>	- Instrucción que ajusta la fecha actual del sistema.
<b>Declare</b>	- Instrucción que hace referencias externas en DLLs.
<b>DefTipo de Dato</b>	- Instrucción utilizada en la sección de declaraciones de los forms y módulos para ajustar los tipos de datos de las variables, constantes y procedimientos Function.
<b>Dim</b>	- Instrucción utilizada en módulos y procedimientos para declarar y asignar espacio de almacenamiento para las variables.
<b>Do...Loop</b>	- Esta estructura repite un bloque de instrucciones contenidas en su interior hasta que una condición sea verdadera o mientras sea verdadera.
<b>End</b>	- Instrucción que da fin a un programa, un procedimiento o un bloque en VB.
<b>Erase</b>	- Instrucción que re-inicializa los elementos de los arreglos.
<b>Err</b>	- Ajusta el código de control de error Err a un valor especificado.
<b>Error</b>	- Instrucción que simula la ocurrencia de un error, tanto de VB como del usuario.
<b>Exit</b>	- Instrucción que sale o da fin a bucles



- 
- Do...Loop, For...Next, o procedimientos Function o Sub.
- For...Next** - Declaraciones utilizadas en conjunto para repetir un grupo de instrucciones, un determinado número de veces.
- Function** - Declara nombres, argumentos y códigos que forman la estructura de instrucciones de un procedimiento.
- Get** - Instrucción que lee un archivo en disco y retorna el contenido de un registro.
- Global** - Utilizada en módulos globales para declarar variables y asignar áreas de almacenamiento en memoria.
- GoSub...Return** - Conjunto de instrucciones que desvían el proceso a una subrutina dentro de un procedimiento, retornando a la instrucción siguiente a la instrucción GoSub.
- GoTo** - Instrucción que desvía incondicionalmente el proceso a una línea especificada dentro de un procedimiento (lo desvía pero sin regresar).
- If...Then...Else** - Conjunto de instrucciones que permiten la ejecución de sentencias que representan condiciones basadas en la evaluación de determinadas expresiones.
- Input#** - Instrucción que lee datos de un archivo secuencial en disco y los asocia a los registros como variables del programa.

- 
- Kill** - Instrucción que elimina los archivos del disco.
- Let** - Instrucción que asocia el valor de una expresión a una variable.
- Line Input#** - Instrucción que lee un registro de un archivo secuencial en disco de una variable de tipo string.
- Load...Unload** - Instrucción que carga una pantalla (form) o un control en memoria.
- Lock...Unlock** - Instrucciones que controlan el acceso de otros procesos a registros específicos o archivos enteros en disco.
- Lset** - Instrucción que justifica a la izquierda un string dentro de una variable tipo string de longitud fija, o copia un registro en otro.
- Mid\$** - Instrucción que extrae una cadena de caracteres (string) desde una posición.
- MkDir** - Instrucción que crea un directorio en el camino o disco especificado.
- MsgBox** - Instrucción que muestra un mensaje en un cuadro de diálogo, cuyo tipo se debe definir.
- Name** - Instrucción que cambia el nombre de un archivo en el disco o de un directorio.
- On Error** - Habilita la rutina de tratamientos de errores y especifica la posición de la misma dentro de un procedimiento.
- On...GoSub y On...GoTo** - Conjunto de instrucciones que desvían el proceso a una de las diversas líneas

especificadas, dependiendo del valor de una expresión.

- Open** - Instrucción que abre un archivo en disco, es decir, permite operaciones de entrada/salida.
- Option Base** - Instrucción que ajusta el índice menor de una matriz.
- Print#** - Instrucción que escribe datos en un archivo secuencial.
- Put** - Instrucción que escribe un registro en un archivo del disco.
- Randomize** - Instrucción que inicializa el generador de números aleatorios.
- ReDim** - Instrucción utilizada en nivel de procedimientos para declarar arreglos dinámicos y asignar o reasignar áreas de almacenamiento para ellos.
- Rem** - Instrucción que indica que la línea a la cual pertenece, es solamente una línea de comentario y no ejecutable.
- Reset** - Cierra los archivos abiertos.
- Resume** - Instrucción que continua la ejecución de un programa después de que una rutina de tratamiento de errores es encontrada.
- Return** - Instrucción que causa un retorno incondicional de una subrutina (GoSub).
- Rmdir** - Elimina el directorio especificado.
- RSet** - Instrucción que ejecuta exactamente lo mismo

- que la instrucción LSet, sólo que para la derecha en lugar de la izquierda, como en esa instrucción.
- SavePicture** - Almacena un diseño de un form o de un control de picture box en un archivo.
- Select Case** - Instrucción que ejecuta una o varias instrucciones, dependiendo del valor de una expresión.
- SendKeys** - Instrucción que envía una o más teclas a la ventana activa, como si hubieran sido escritas vía teclado.
- Static** - Instrucción utilizada en el nivel de procedimientos para declarar variables estáticas y asignar espacio en memoria para ellas.
- Stop** - Detiene la ejecución de un programa y abre la ventana de códigos.
- Sub** - Declara el nombre, los argumentos y códigos que forman la estructura de un procedimiento Sub.
- Time\$** - Instrucción que ajusta la hora del sistema en la memoria, pero no en la RAM CMOS, y solamente si una máquina está conectada.
- Type** - Instrucción utilizada en módulos globales para definir tipos de datos que contienen uno o más elementos.
- While...Wend** - Conjunto que ejecuta una serie de instrucciones en bucle, cuando una determinada condición es verdadera.

- Width#** - Instrucción que asocia la longitud de una línea de salida a un archivo.
- Write#** - Instrucción que asocia la longitud de una línea de salida a un archivo.

\* **Funciones en Visual Basic.**- Las funciones son operaciones unarias efectuadas con una expresión que retornan un resultado.

Así tenemos:

- Abs** - Calcula el valor absoluto de una expresión numérica de cualquier tipo.
- Asc** - Da por resultado un valor numérico que es el código ANSI de un caracter.
- Atn** - Calcula el arcotangente de una expresión numérica.
- CCur** - Convierte una expresión numérica en un valor tipo Currency.
- CDbl** - Convierte una expresión numérica en un valor de doble precisión.
- Chr\$** - Da por resultado el caracter ASCII de un código.
- CInt** - Convierte una expresión numérica en un valor Integer.
- CLng** - Convierte una expresión numérica en un valor tipo Long.
- Command\$** - Da por resultado una función string, conteniendo una parte del argumento de la línea de mandato utilizada para ejecutar VB, o por una aplicación en VB.
- Cos** - Calcula el coseno de un ángulo expresado en radianes.

- 
- Csng** - Convierte una expresión numérica en un valor de simple precisión.
- CurDir\$** - Retorna el directorio y el path en uso para la unidad especificada.
- Date\$** - Da por resultado la fecha actual del sistema, en un string de 10 caracteres, con el siguiente formato: mm-dd-aaaa.
- DateSerial** - Da por resultado un número único que representa la fecha de los argumentos.
- DateValue** - Da por resultado un número único que representa la fecha del argumento.
- Day** - Da por resultado un número entero entre 1 y 31, que representa el día del mes correspondiente al valor del argumento.
- Dir\$** - Retoma el nombre del primer archivo que satisface el modelo especificado.
- DoEvents** - Permite que otro procesamiento (si existe en el entorno de operación) sea ejecutado junto con la aplicación de VB.
- Environ\$** - Da por resultado un string del entorno de operación del sistema.
- EOF** - Da por resultado un valor que indica si se ha alcanzado o no el fin de archivo, durante una operación de lectura.
- Err, Err1** - Funciones que retornan el estado de error que puede haber ocurrido durante la ejecución de las aplicaciones en VB.

- 
- Error\$** - Retorna el mensaje de error correspondiente al código de error ocurrido.
- Exp** - Calcula el número  $e$  elevado a una potencia.
- FileAttr** - Retorna informaciones referentes al archivo abierto, del tipo atributos.
- Fix** - Convierte una expresión numérica en su parte entera, simplemente truncando su parte fraccionaria.
- Format\$** - Formatea la salida de un valor, para efectos visuales.
- FreeFile** - Retorna el siguiente número disponible para la transacción de archivos.
- Hex\$** - Retorna un string alfanumérico, que representa el valor hexadecimal, del argumento expresado en el sistema decimal.
- Hour** - Retorna un valor entero, que representa la hora del día correspondiente al argumento numérico.
- Input\$** - Retorna un string de caracteres leídos de un archivo.
- InputBox\$** - Muestra un cursor (prompt) en un cuadro de diálogo, en espera que el usuario escriba algo o seleccione un botón, y retorna al contenido del cuadro de edición.
- InStr** - Retorna la posición de la primera ocurrencia de un carácter de un string dentro de otra.
- Int** - Convierte una expresión numérica al mayor valor entero que sea menor o igual a su valor.
- LBound** - Retorna el menor subíndice disponible para la dimensión indicada de una determinada matriz.

- 
- LCase\$** - Retorna un string de caracteres en donde todas las letras han sido convertidas en mayúsculas.
- Left\$** - Retorna un string que consta de los caracteres que se encuentran más a la izquierda de la cadena especificada.
- Len** - Retorna la longitud de la cadena de caracteres especificada en número de caracteres, o el número de bytes de almacenamiento requerido por una variable.
- LoadPicture** - Carga un diseño en un form o en un control picture box.
- Loc** - Retorna la posición actual dentro de un archivo abierto.
- LOF** - Retorna el tamaño de un archivo abierto en bytes.
- Log** - Calcula el logaritmo de una expresión numérica.
- LTrim\$** - Retorna el string dado como argumento, ajustado a la izquierda.
- Mid\$** - Retorna un string, que es parte de otra función especificada en la sintaxis, con un número especificado de caracteres.
- Minute** - Da por resultado un valor entero comprendido entre 0 y 59, que representa el minuto de la hora correspondiente al argumento.
- Month** - Da por resultado un entero entre 1 y 12, que representa el mes del año correspondiente al valor del argumento.
- MsgBox** - Muestra un mensaje en el cuadro de diálogo y espera que el usuario elija un botón de opción, colocando



además un valor que indica el botón que ha sido seleccionado.

- Now** - Da por resultado un número que representa la fecha y la hora actual, según el valor por defecto o la forma de representación de esos valores en el sistema que se está utilizando.
- Oct\$** - Retorna un string alfanumérico que representa el valor octal del argumento expresado en base decimal.
- QBColor** - Retorna el código de colores RGB correspondiente al número del color.
- RGB** - Da por resultado un valor correspondiente a los valores de los colores RGB.
- Right\$** - Retorna una cadena de caracteres consistentes en caracteres que se encuentran más a la derecha.
- Rnd** - Da por resultado un número aleatorio del tipo simple precisión, comprendido entre 0 y 1.
- RTrim\$** - Retorna una copia de la expresión de cadena especificada con los espacios de la derecha suprimidos.
- Second** - Da por resultado un valor entero, comprendido entre 0 y 59, que representa el segundo del minuto correspondiente al argumento.
- Seek** - Retorna la posición del archivo abierto.
- Sgn** - Da por resultado un valor, indicando el signo de una expresión numérica.
- Shell** - Ejecuta un programa, siempre que éste pueda ser ejecutado.
- Sin** - Retorna el seno de un ángulo expresado en radianes.

- 
- Space\$** - Da por resultado una expresión string que consiste en un determinado número de espacios en blanco.
- Spc** - Salta un determinado número de espacios en blanco desde la posición de impresión actual, en una instrucción Print # o en un método Print.
- Sqr** - Calcula la raíz cuadrada de una expresión numérica de valor positivo.
- Str\$** - Da por resultado una representación alfanumérica del valor de una expresión numérica.
- Tab** - Mueve el cursor de texto a una posición de impresión especificada, cuando se utiliza en una instrucción Print # o con el método Print.
- Tan** - Calcula la tangente de un ángulo expresado en radianes.
- Time\$** - Función que retorna la hora actual del sistema.
- Timer** - Da por resultado el número de segundos transcurridos desde la medianoche.
- TimeValue** - Da por resultado un número que representa la hora del argumento.
- UBound** - Da por resultado el mayor subíndice disponible para la dimensión indicada de una determinada matriz.
- UCase\$** - Da por resultado una cadena, en la cual todas las letras deben ser convertidas en mayúsculas.
- Val** - Da por resultado el valor numérico de una cadena de caracteres alfanuméricos.
- WeekDay** - Da por resultado un número entero comprendido entre 1 (equivalente a Sunday-domingo), y 7 (equivalente a

Saturday-sábado), que representa el día de la semana.

**Year** - Da por resultado un número entero comprendido entre 1753 y 2078, que representa el año correspondiente al argumento.

\* **Métodos en Visual Basic.**- Soporta la mayoría de las instrucciones y funciones de otras versiones de Basic. Además añade algunas nuevas características en forma de métodos.

Un método en VB es una palabra reserva, similar a una función o instrucción, que siempre actúa sobre un objeto en particular. Para cada objeto existe un conjunto predefinido de métodos que puede utilizar.

A continuación tenemos algunos métodos de interés, así:

- AddItem** - Añade un ítem en un control List Box o Combo Box, durante la ejecución de una aplicación.
- Circle** - Utilizado para trazar círculos, elipses o arcos.
- Clear** - Utilizado para limpiar el contenido del Clipboard de Windows.
- Cls** Utilizado para limpiara un form o un picture box.
- Drag** - Inicia, termina o cancela el movimiento de controles.
- EndDoc** - Cierra el envío de un documento a la impresora.
- GetData** - Retorna un diseño o gráfico del Clipboard.
- GetFormat** - Da por resultado un valor entero, indicando si existe algo en el Clipboard de Windows que corresponda al formato especificado.
- Line** - Utilizado para diseñar líneas y rectángulos en una

pantalla o en un control picture box o en la impresión.

**LinkExecute** - Envía mandatos a otra aplicación en una conversación DDE (Dynamic Data Exchange).

**LinkPoke** - Transfiere el contenido de un control a una aplicación en un servidor DDE.

**LinkRequest** - Solicita al servidor de una conversación DDE que actualice el contenido de un control.

**LinkSend** - Transfiere el contenido de un control picture box a la aplicación cliente de una conversación DDE.

**Move** - Desplaza una pantalla o un control.

**New Page** - Utilizado para cerrar la página actual y avanzar a la próxima.

**Point** - Retorna el color RGB del punto especificado en una pantalla o control picture box.

**PrintForm** - Envía una imagen bit a bit de un form a la impresión.

**Print** - Imprime un texto en un form o en un control picture box o en un objeto printer, utilizando el color y el tipo de letra actual.

**PSet** - Utilizado para diseñar puntos en una pantalla o en un control picture box o en la impresión, con un color específico.

**RemoveItem** - Elimina un ítem de un list Box o combo box durante la ejecución de un programa.

**Scale** - Define el sistema de coordenadas para una pantalla o un control picture box u object printer.

- 
- SetFocus** - Ajusta el centro activo en un determinado objeto.
- SetText** - Coloca en el Clipboard una expresión alfanumérica (string) utilizando un formato propio del Clipboard especificado en la sintaxis.
- SetData** - Coloca en el Clipboard un diseño, utilizando un formato propio del mismo especificado en la sintaxis.
- Show** - Muestra un form.
- TextHeight** - Retorna la altura de una expresión string, en el caso de que fuese impresa con el tipo de letra actual de una pantalla o de un control picture box u object printer.
- TextWidth** - Retorna la anchura de una expresión string, en el caso de que fuese impresa con el tipo de letra actual de una pantalla o de un control picture box u object printer.

```
Sub Command1_Click ()
'PROCEDIMIENTO PARA REALIZAR UN INGRESO DE DATOS EN UN SISTEMA POR COLA
LOS DATOS SERAN ALMACENADOS EN LA TABLA SIS_COLA
```

```
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_cola")
iniciov = form13.MaskedEdit1.Text
Do While dbnaset.EOF = False
    If Text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            dbnaset.Edit
            dbnaset("nbidsq") = Text6.Text
            dbnaset("nsaq") = Text5.Text
            dbnaset("anblo") = Text7.Text
            dbnaset("nhtime") = Text8.Text
            dbnaset("mqtime") = Text9.Text
            dbnaset.Update
        End If
    End If
    dbnaset.MoveNext
Loop
MsgBox ("DATOS GRABADOS")
form19.Show
```

```
End Sub
```

```
Sub Command2_Click ()
form1.Show
Unload form12
End Sub
```

```
Sub Command3_Click ()
form1.Show
form12.Hide
tex1 = ""
tex2 = ""
tex3 = ""
tex4 = ""
tex5 = ""
tex6 = ""
tex7 = ""
tex8 = ""
tex9 = ""
End Sub
```

```
Sub Text1_KeyPress (keyascii As Integer)
If keyascii < Asc("a") Or keyascii > Asc("z") Then
    Beep
    keyascii = 0
    Text1.Text = ""
End If
```

```
End Sub
```

```
Sub Text1_LostFocus ()
Text1.Text = UCase$(Text1.Text)
Dim db As database
```

```

Dim dbnaset As dynaset
Dim bet As Variant
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_cola")
iniciov = form13.MaskedEdit1.Text
ban = 1
bet = 0
Do While dbnaset.EOF = False

    If Text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            ban = 0
            If dbnaset("nsaq") = 0 Then
                Text2.Text = dbnaset("traff")
                Text3.Text = dbnaset("nbids")
                Text4.Text = dbnaset("ndv")
                Text5.SetFocus
            Else
                bet = 1
            End If
        End If
    End If
    dbnaset.MoveNext
Loop

If ban = 1 Then
    Text1.Text = ""
    MsgBox ("DATOS INCOMPLETOS")
End If

If bet = 1 Then
    Text1.Text = ""
    MsgBox ("ESTOS DATOS ESTAN YA REGISTRADOS")
End If

End Sub

Sub Text2_KeyPress (keyascii As Integer)
Static puntodecimalusado As Integer
Dim ayuda As Variant
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii > Asc("A") And keyascii < Asc("Z") Then
    Beep
    keyascii = 0
    Text2.Text = ""
Else
    Select Case keyascii
    Case Asc("0") To Asc("9")
        ayuda = ayuda + Chr$(keyascii)
    Case Asc(".")
        If puntodecimalusado Then
            keyascii = 0
            Beep
        Else
            puntodecimalusado = True
            ayuda = ayuda + Chr$(keyascii)
        End If
    End Select
End Sub

```

```
        Text2.Text = ayuda
    Case Asc(",")
        ' coma no afecta
    Case Else
        keyascii = 0
        Beep
End Select

End If

End Sub

Sub Text3_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    Text3.Text = ""
End If

End Sub

Sub Text4_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    Text4.Text = ""
End If

End Sub

Sub Text5_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
    Beep
    keyascii = 0
    Text5.Text = ""
End If

End Sub

Sub Text6_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
    Beep
    keyascii = 0
    Text6.Text = ""
End If

End Sub

Sub Text7_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
    Beep
    keyascii = 0
    Text7.Text = ""
End If
```



End Sub

Sub Text8\_KeyPress (keyascii As Integer)

If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii < Asc("Z") Then

    Beep

    keyascii = 0

    Text8.Text = ""

End If

End Sub

Sub Text9\_KeyPress (keyascii As Integer)

If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii < Asc("Z") Then

    Beep

    keyascii = 0

    Text9.Text = ""

End If

End Sub

Sub Text9\_LostFocus ()

command1.Enabled = True

End Sub

```

Sub BLOQ_Click ()
*****PROCEDIMIENTO PARA OBTENER EL GRAFICO DE CIRCUITOS BIDIRECCIONALES
BLOQUEADOS EN UNA HORA PICO
Dim db As database
Dim dnaset As dynaset
Dim v As Variant
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("trafico_saliente") 'LOS DATOS SE OBTENDRAN DE LA TABLA
TRAFICO SALIENTE
iniciov = form13.MaskedEdit1.Text 'TENEMOS LA FECHA DE TOMA DE DATOS
  Do While dnaset.EOF = False
    If iniciov = dnaset("inicio") Then
      'DAMOS VALORES A LOS PARAMETROS PARA OBTENER LOS GRAFICOS
      form8.Graph1.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
      form8.Graph1.GraphData = dnaset("nblov")
    End If
    dnaset.MoveNext
  Loop

form8.Show 'DAMOS PASO A LA VIZUALIZACION DEL GRAFICO EL MISMO QUE SE
ENCUENTRA EN LA FORMA 8.
form1.Hide

```

End Sub

```

Sub CONS_Click ()
*****REALIZA EL GRAFICO DE TRAFICO SALIENTE EN HORA PICO*****
Dim db As database
Dim dnaset As dynaset
Dim v As Variant
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("trafico_saliente")
iniciov = form13.MaskedEdit1.Text 'DETERMINA LA FECHA PERIODO DE LA TOMA DE DATOS
Dim var As Variant
Do While dnaset.EOF = False
  If iniciov = dnaset("inicio") Then
    var = dnaset("horap")
    'DAMOS VALORES A LOS PARAMETROS REQUERIDOS PARA LA GRAFICA
    form5.Graph1.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
    form5.Graph1.GraphData = dnaset("traff")
  End If
  dnaset.MoveNext
Loop
form5.Text2 = var
form5.Show 'INSTRUCCIONES PARA LA COMUNICACION ENTRE FORMAS.
form1.Hide

```

End Sub

```

Sub CONS3_Click ()
'UTILIZAMOS LA FORMA 40 PARA INGRESAR
'LA CENTRAL DE LA QUE SE TOMARAN LOS DATOS Y DETERMINAR
'LA CALIDAD DE SERVICIO
form40.Panel3D1.Visible = False
form40.Command2.Visible = False
form40.Command3.Visible = False

```

```
form7.Command2.Visible = False
form7.Command3.Visible = False
form40.Show
```

```
End Sub
```

```
Sub CONS4_Click ()
Dim db As database
Dim dnaset As dynaset
Dim var As Variant
Dim i As Integer
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("mtp_trafico")
var = InputBox("Ingrese por favor la Central")
Do While dnaset.EOF = False
    If var = dnaset("ruta") Then
        form8.Graph1.GraphData = dnaset("nblov")
    End If
    dnaset.MoveNext
Loop
form8.Visible = True
form1.Visible = False
End Sub
```

```
Sub DATOS_Click ()
Unload form20
form20.Show
form1.Hide
```

```
End Sub
```

```
Sub DATOS3_Click ()
Unload form33
form33.Show
form1.Hide
End Sub
```

```
Sub DATOS4_Click ()
form35.Show
form1.Hide
```

```
End Sub
```

```
Sub DP_Click ()
form25.Show
form1.Hide
End Sub
```

```
Sub DPC_Click ()
PROCEDIMIENTO PARA TENER LA GRAFICA DE CARGA DEL PROCESADOR EN DIA PICO.
TOMANDO
EN CUENTA UNICAMENTE LAS MEDICIONES HECHAS EN CADA HORA DEL DIA.
Dim db As database
Dim dnaset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("carga_procesador") 'ESTOS DATOS ESTAN ALMACENADOS EN LA
TABLA CARGA_PROCESADOR
Dim var As Variant
iniciov = form13.MaskedEdit1.Text
```

```

Do While dnaset.EOF = False
  If iniciov = dnaset("inicio") Then
    var = dnaset("diap")
    form2.Graph1.ThisPoint = Val(Mid$(dnaset("hora"), 1, 2))
    form2.Graph1.GraphData = dnaset("pload")
  End If
  dnaset.MoveNext
Loop
form2.Text2.Text = var
form2.Show

```

```
End Sub
```

```

Sub DTF_Click ()
Unload form31
Unload form38

```

```

'Unload form1
Unload form2
Unload form3
Unload form4
Unload form5
Unload form6
Unload form7
Unload form8
Unload form9
Unload form11
Unload form12
'Unload form13

```

```

Unload form15
Unload form16
Unload form17
Unload form18

```

INSTRUCCIONES NECESARIAS PARA LA ELABORACION DE LOS  
'CUADROS ESTADISTICOS, EN EL QUE SE DETALLARAN, DATOS COMO:  
'NOMBRE DE LA CENTRAL, CIRCUITOS BIDIREC. BLOQUEADOS EN HP,  
'CARGA DE PROCESADOR EN HORA PICO ETC.

```

Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
'Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

```

```

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

```

```
Do While dnaset.EOF = False
  If iniciov = dnaset("inicio") Then
    var = dnaset("diap")
    'ELABORACION DE LA GRAFICA
    form2.Graph1.ThisPoint = Val(Mid$(dnaset("hora"), 1, 2))
    form2.Graph1.GraphData = dnaset("pload")
  End If
  dnaset.MoveNext
Loop
form2.Text2.Text = var
form2.Show

End Sub

Sub DTF_Click ()
Unload form31
Unload form38

Unload form1
Unload form2
Unload form3
Unload form4
Unload form5
Unload form6
Unload form7
Unload form8
Unload form9
Unload form11
Unload form12
Unload form13

Unload form15
Unload form16
Unload form17
Unload form18
```

```

    i = i + 1
    dat.MoveNext
    Loop
    form10.Text90.Text = 1
    form10.Text82.Text = 2
    form10.Text83.Text = 3
    form10.Text84.Text = 4
    form10.Text85.Text = 5
    form10.Text86.Text = 6
    form10.Text87.Text = 7
    form10.Text88.Text = 8
    form10.Text89.Text = 9

```

```

form10.Text1.Text = VEC(0)
form10.Text2.Text = VEC(1)
form10.Text3.Text = VEC(2)
form10.Text4.Text = VEC(3)
form10.Text5.Text = VEC(4)
form10.Text6.Text = VEC(5)
form10.Text7.Text = VEC(6)
form10.Text8.Text = VEC(7)
form10.Text11.Text = VEC(8)

```

```

form10.Text12.Text = abr(0)
form10.Text13.Text = abr(1)
form10.Text14.Text = abr(2)
form10.Text15.Text = abr(3)
form10.Text16.Text = abr(4)
form10.Text17.Text = abr(5)
form10.Text18.Text = abr(6)
form10.Text10.Text = abr(7)
form10.Text9.Text = abr(8)
'DATOS PARA LA FORMA 43

```

```

i = 0
'BLOQUE PARA HALLAR EL VALOR DE CIRCUITOS BIDIRECCIONALES POR RUTA
'LOS QUE ESTAN EN LA TABLA SIS_PERDIDA"
Set dat = db.CreateDynaset("sis_perdida")
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Do While dat.EOF = False

    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            var(i) = dat("ndv")
            i = i + 1
        End If
    End If
    dat.MoveNext
Loop

```

```

ReDim bloq(35) As Variant
Set dat = db.CreateDynaset("sis_perdida")
i = 0
Do While dat.EOF = False
    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            bloq(i) = dat("anblo")
            i = i + 1
        End If
    End If
    dat.MoveNext

```

Loop

```

form10.Text27.Text = var(0)
form10.Text19.Text = var(1)
form10.Text20.Text = var(2)
form10.Text21.Text = var(3)
form10.Text22.Text = var(4)
form10.Text23.Text = var(5)
form10.Text24.Text = var(6)
form10.Text25.Text = var(7)
form10.Text26.Text = var(8)
'FINAL DEL PRIMER BLOQUE
'SEGUNDO BLOQUE
form10.Text37.Text = bloq(0)
form10.Text38.Text = bloq(1)
form10.Text39.Text = bloq(2)
form10.Text44.Text = bloq(3)
form10.Text45.Text = bloq(4)
form10.Text40.Text = bloq(5)
form10.Text46.Text = bloq(6)
form10.Text47.Text = bloq(7)
form10.Text48.Text = bloq(8)

```

instrucciones para hallar el trafico los valores del trafico saliente  
y entrante y determinar la suma

```

Set dat = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While dat.EOF = False
    If iniciov = dat("inicio") Then

        If abr(i) + "O" = dat("ruta") Then
            saliente(i) = dat("traff")
            i = i + 1
        End If

        If abr(j) + "I" = dat("ruta") Then
            entrante(j) = dat("traff")
            j = j + 1
        End If
    End If
    dat.MoveNext

```

Loop

```
'instrucciones para hallar el total
For i = 0 To 35
    total(i) = saliente(i) + entrante(i)
Next i
```

'bloque para visualizar los valores de trafico entrante y saliente

```
form10.Text42.Text = entrante(0)
form10.Text43.Text = entrante(1)
form10.Text49.Text = entrante(2)
form10.Text50.Text = entrante(3)
form10.Text51.Text = entrante(4)
form10.Text52.Text = entrante(5)
form10.Text53.Text = entrante(6)
form10.Text54.Text = entrante(7)
form10.Text55.Text = entrante(8)
```

```
form10.Text63.Text = saliente(0)
form10.Text62.Text = saliente(1)
form10.Text61.Text = saliente(2)
form10.Text60.Text = saliente(3)
form10.Text59.Text = saliente(4)
form10.Text58.Text = saliente(5)
form10.Text57.Text = saliente(6)
form10.Text56.Text = saliente(7)
form10.Text41.Text = saliente(8)
```

```
form10.Text65.Text = total(0)
form10.Text64.Text = total(1)
form10.Text74.Text = total(2)
form10.Text75.Text = total(3)
form10.Text76.Text = total(4)
form10.Text77.Text = total(5)
form10.Text78.Text = total(6)
form10.Text79.Text = total(7)
form10.Text80.Text = total(8)
```

'halla el valor de porcentaje de congestión de tráfico saliente

```
Set dat = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While dat.EOF = False
    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            cong(i) = dat("ccong")
            i = i + 1
        End If
    End If
    dat.MoveNext
```

Loop

```
form10.Text81.Text = cong(0)
```



```

form10.Text66.Text = cong(1)
form10.Text67.Text = cong(2)
form10.Text68.Text = cong(3)
form10.Text69.Text = cong(4)
form10.Text70.Text = cong(5)
form10.Text71.Text = cong(6)
form10.Text72.Text = cong(7)
form10.Text73.Text = cong(8)

```

```

form10.Show
form1.Hide
End Sub

```

```
Sub ENT_Click ()
```

```
End Sub
```

```
Sub ENTSEG_Click ()
```

```
'ELABORACION DE LA GRAFICA DE LLAMADAS ENTRANTES POR SEGUNDO EN HORA  
PICO
```

```
'DATOS TOMADOS EN UNA HORA ESPECIFICA TODOS LOS DIAS DEL MES.
```

```
'ESTOS DATOS SE ENCUENTRAN EN ALMACENADOS EN LA TABLA
```

```
CARGA_PROCESADOR_DIA
```

```
Dim db As database
```

```
Dim dset As dynaset
```

```
Dim var As Variant
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
Set dset = db.CreateDynaset("carga_procesador_dia")
```

```
iniciov = form13.MaskedEdit1.Text 'DETERMINAMOS LA FECHA DE TOMA DE DATOS.
```

```
Do While dset.EOF = False
```

```
    If iniciov = dset("inicio") Then
```

```
        var = dset("horap")
```

```
        form6.Graph1.ThisPoint = Val(Mid$(dset("dia"), 1, 2)) 'DAMOS VALORES A LOS
```

```
PARAMETROS
```

```
        form6.Graph1.GraphData = dset("fchdi")
```

```
        'PARA LA ELABORACION DEL
```

```
GRAFICO.
```

```
        End If
```

```
        dset.MoveNext
```

```
Loop
```

```
form6.Text2 = var
```

```
form6.Show
```

```
form1.Hide
```

```
End Sub
```

```
Sub Form_Load ()
```

```
Dim db As database
```

```
Dim naset As dynaset
```

```
Dim ncola As dynaset
```

```
Dim nperdida As dynaset
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
'Set naset = db.CreateDynaset("fecha")
```

```
Set ncola = db.CreateDynaset("sis_cola")
```

```
Set nperdida = db.CreateDynaset("sis_perdida")
```

```
Dim x1 As Variant
```

```
Dim x2 As Variant
```

```
Dim x3 As Variant
```

```
Dim x4 As Variant
```

```

Dim variable As Variant
variable = Date$
x1 = Mid$(variable, 1, 2)
x2 = Mid$(variable, 4, 2)
form13.MaskedEdit1.Text = "01-05-1995"
form13.MaskedEdit2.Text = "31-05-1995"
form13.MaskedEdit3.Text = "25-05-1995"
form13.MaskedEdit4.Text = "09:30 am a 10:30 am"

```

```

Do While ncola.EOF = False
    If ncola("final") <> "" Then
        x3 = Mid$(ncola("final"), 1, 2)
        x4 = Mid$(ncola("final"), 4, 2)
        If Val(x1) > Val(x3) Then
            If Val(x1) - Val(x3) = 1 Then
                form13.Show
                form1.Hide
            End If
        End If
    End If
    ncola.MoveNext
    Loop
form13.Show
form1.Hide
End Sub

```

```

Sub GIT_Click ()
Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("Sis_Perdida")
Set dat = db.CreateDynaset("Datos")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
inicio = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

```

```

Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant

```

```

ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
  If iniciov = naset("inicio") Then

    If abr(i) + "O" = naset("ruta") Then
      saliente(i) = naset("traff")
      i = i + 1
    End If

    If abr(j) + "I" = naset("ruta") Then
      entrante(j) = naset("traff")
      j = j + 1
    End If
  End If
  naset.MoveNext
Loop
Dim tot As Variant

tot = 0
For i = 0 To 35
  total(i) = saliente(i) + entrante(i)
  tot = tot + total(i)
Next i
Dim bet As Variant
Dim bet1 As Variant
bet = total(0) + total(3) + total(5) + total(12) + total(17) + total(20) + total(21) + total(22)
bet = bet + total(23) + total(24) + total(25)

bet1 = total(1) + total(2) + total(4) + total(7) + total(8) + total(9) + total(10) + total(11)
bet1 = bet + total(13) + total(14) + total(15) + total(16) + total(18) + total(19)

form4.Graph1.GraphData = bet1 * 100 / tot
form4.Graph1.GraphData = bet * 100 / tot
form4.Graph1.GraphData = total(15) * 100 / tot
form4.Graph1.GraphData = total(26) * 100 / tot
form4.Graph1.GraphData = total(27) * 100 / tot
form4.Graph1.GraphData = total(28) * 100 / tot
form4.Graph1.GraphData = total(29) * 100 / tot
form4.Graph1.GraphData = total(30) * 100 / tot
form4.Graph1.GraphData = total(31) * 100 / tot

form4.Graph1.LegendText = "Quito"
form4.Graph1.LegendText = "Area 2"
form4.Graph1.LegendText = "Tel. Movil"
form4.Graph1.LegendText = "Ambato Transito"
form4.Graph1.LegendText = "Cuenca Transito"
form4.Graph1.LegendText = "Guayaquil Transito"
form4.Graph1.LegendText = "Ibarra Transito"
form4.Graph1.LegendText = "Quito Transito 1"
form4.Graph1.LegendText = "Internacional"

form4.Show
form1.Hide

```

```
End Sub
```

```
Sub GR_Click ()
```

```
'AQUI SE REALIZAN LAS GRAFICAS, DE LAS MEDICIONES PRINCIPALES
```

```
'DEL PROCESADOR QTS2
```

```
'SE PRESENTARAN EN FORMA CONSECUTIVA CADA UNA
```

```
Dim bd As database
```

```
Dim dnaset As dynaset
```

```
Dim var As Variant
```

```
Dim v As Variant
```

```
Dim i As Integer
```

```
Dim ban As Integer
```

```
Set bd = OpenDatabase("c:\vb\pruebas.mdb")
```

```
Set dnaset = bd.CreateDynaset("carga_procesador_dia")
```

```
iniciov = form13.MaskedEdit1.Text
```

```
Do While dnaset.EOF = False
```

```
    If iniciov = dnaset("inicio") Then
```

```
        var = dnaset("horap")
```

```
        form9.Graph2.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
```

```
        form9.Graph2.GraphData = dnaset("pload")
```

```
    End If
```

```
        dnaset.MoveNext
```

```
Loop
```

```
*****GRAFICO DEL CARGA DE PROCESADOR EN HORA PICO
```

```
Dim db As database
```

```
Dim dyns As dynaset
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
Set dyns = db.CreateDynaset("carga_procesador")
```

```
iniciov = form13.MaskedEdit1.Text
```

```
Do While dyns.EOF = False
```

```
    If iniciov = dyns("inicio") Then
```

```
        form9.Graph5.ThisPoint = Val(Mid$(dyns("hora"), 1, 2))
```

```
        form9.Graph5.GraphData = dyns("pload")
```

```
    End If
```

```
        dyns.MoveNext
```

```
Loop
```

```
form9.Show "VISUALIZACION DE LAS FORMAS CON LOS GRAFICOS
```

```
form1.Hide
```

```
End Sub
```

```
Sub HP_Click ()
```

```
form28.Show
```

```
form1.Hide
```

```
End Sub
```

```
Sub HPC_Click ()
```

```
'OBTENCION DEL GRAFICO DE CARGA DE PROCESADOR EN HORA PICO
```

```
Dim bd As database
```

```
Dim dnaset As dynaset
```

```
Dim var As Variant
```

```
Dim v As Variant
```

```
Dim i As Integer
```

```

Dim ban As Integer
Set bd = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = bd.CreateDynaset("carga_procesador_dia") 'ABRE LA TABLA UTILIZAR SUS DATOS
EN LA GRAFICA.
iniciov = form13.MaskedEdit1.Text 'PONEMOS LA FECHA DE TOMA DE DATOS.

Do While dnaset.EOF = False
    If iniciov = dnaset("inicio") Then
        var = dnaset("horap") 'ELABORACION DE LAS GRAFICAS
        form3.Graph1.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
        form3.Graph1.GraphData = dnaset("pload")
    End If
    dnaset.MoveNext
Loop

form3.Text2.Text = var
dnaset.Close 'CIERRE DE LA BASE DE DATOS Y DYNASETS
bd.Close
form3.Show
End Sub

Sub PD_Click ()
form15.Text9.Text = "POR DESTINO"
form15.Show
form1.Hide

End Sub

Sub PHP_Click ()
Unload form10
INSTRUCCIONES PARA LA ELABORACION DE LOS CUADROS EN LOS QUE ESTAN LA
CARGA DE
PROCESADOR EN DIA Y HORA PICO, ASI COMO LAS LLAMADAS ENTRANTES POR
SEGUNDO EN
HORA PICO, DATOS QUE SERAN MODIFICADOS PARA OBSERVAR CAMBIOS EN LAS
GRAFICAS.
Dim db As database
Dim naset As dynaset
Dim saliente As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("carga_procesador_dia")
Set saliente = db.CreateDynaset("trafico_Saliente")
iniciov = form13.MaskedEdit1.Text
Dim i As Integer
i = 0
Dim var As Variant
Dim fin As Variant
ReDim dia(31) As Variant
ReDim plod(31) As Variant
ReDim llam(31) As Variant
ReDim traff(31) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        fin = naset("final")

        plod(i) = naset("pload")
        llam(i) = naset("fchdi")
        i = i + 1
    End If

```

```

naset.MoveNext
Loop

i = 0

Do While saliente.EOF = False
  If iniciov = saliente("inicio") Then
    traf(i) = saliente("traff")
    dia(i) = Val(Mid$(saliente("dia"), 1, 2))
    i = i + 1
  End If
  saliente.MoveNext

Loop

form31.Label6.Caption = iniciov
form31.Label7.Caption = fin

'DAMOS VALORES A CADA UNO DE LOS CUADROS DE TEXTO
form31.Text4.Text = dia(0)
form31.Text5.Text = dia(1)
form31.Text6.Text = dia(2)
form31.Text7.Text = dia(3)
form31.Text8.Text = dia(4)
form31.Text9.Text = dia(5)
form31.Text10.Text = dia(6)
form31.Text11.Text = dia(7)
form31.Text12.Text = dia(8)
form31.Text13.Text = dia(9)
form31.Text14.Text = dia(10)
form31.Text15.Text = dia(11)
form31.Text16.Text = dia(12)
form31.Text17.Text = dia(13)

form31.Text18.Text = plod(0)
form31.Text19.Text = plod(1)
form31.Text20.Text = plod(2)
form31.Text21.Text = plod(3)
form31.Text22.Text = plod(4)
form31.Text23.Text = plod(5)
form31.Text24.Text = plod(6)
form31.Text25.Text = plod(7)
form31.Text26.Text = plod(8)
form31.Text27.Text = plod(9)
form31.Text28.Text = plod(10)
form31.Text29.Text = plod(11)
form31.Text30.Text = plod(12)
form31.Text31.Text = plod(13)

form31.Text39.Text = llam(0)
form31.Text40.Text = llam(1)
form31.Text41.Text = llam(2)
form31.Text42.Text = llam(3)
form31.Text43.Text = llam(4)

```

```
form31.Text44.Text = llam(5)
form31.Text45.Text = llam(6)
form31.Text46.Text = llam(7)
form31.Text47.Text = llam(8)
form31.Text48.Text = llam(9)
form31.Text49.Text = llam(10)
form31.Text50.Text = llam(11)
form31.Text51.Text = llam(12)
form31.Text52.Text = llam(13)
```

```
form31.Text56.Text = traf(0)
form31.Text53.Text = traf(1)
form31.Text54.Text = traf(2)
form31.Text55.Text = traf(3)
form31.Text32.Text = traf(4)
form31.Text33.Text = traf(5)
form31.Text34.Text = traf(6)
form31.Text35.Text = traf(7)
form31.Text36.Text = traf(8)
form31.Text37.Text = traf(9)
form31.Text38.Text = traf(10)
form31.Text1.Text = traf(11)
form31.Text2.Text = traf(12)
form31.Text3.Text = traf(13)
```

'datos en el formulario 38

```
form38.Text1.Text = dia(14)
form38.Text2.Text = dia(15)
form38.Text3.Text = dia(16)
form38.Text4.Text = dia(17)
form38.Text5.Text = dia(18)
form38.Text6.Text = dia(19)
form38.Text7.Text = dia(20)
form38.Text8.Text = dia(21)
form38.Text9.Text = dia(22)
form38.Text10.Text = dia(23)
form38.Text11.Text = dia(24)
form38.Text12.Text = dia(25)
form38.Text13.Text = dia(26)
form38.Text14.Text = dia(27)
form38.Text15.Text = dia(28)
form38.Text16.Text = dia(29)
form38.Text17.Text = dia(30)
```

```
form38.Text18.Text = plod(14)
form38.Text19.Text = plod(15)
form38.Text20.Text = plod(16)
form38.Text21.Text = plod(17)
form38.Text22.Text = plod(18)
form38.Text23.Text = plod(19)
form38.Text24.Text = plod(20)
form38.Text25.Text = plod(21)
form38.Text26.Text = plod(22)
```

```
form38.Text27.Text = plod(23)
form38.Text28.Text = plod(24)
form38.Text29.Text = plod(25)
form38.Text30.Text = plod(26)
form38.Text31.Text = plod(27)
form38.Text32.Text = plod(28)
form38.Text33.Text = plod(29)
form38.Text34.Text = plod(30)
```

```
form38.Text52.Text = llam(14)
form38.Text53.Text = llam(15)
form38.Text54.Text = llam(16)
form38.Text55.Text = llam(17)
form38.Text56.Text = llam(18)
form38.Text57.Text = llam(19)
form38.Text58.Text = llam(20)
form38.Text59.Text = llam(21)
form38.Text60.Text = llam(22)
form38.Text61.Text = llam(23)
form38.Text62.Text = llam(24)
form38.Text63.Text = llam(25)
form38.Text64.Text = llam(26)
form38.Text65.Text = llam(27)
form38.Text66.Text = llam(28)
form38.Text67.Text = llam(29)
form38.Text68.Text = llam(30)
```

```
form38.Text35.Text = traf(14)
form38.Text36.Text = traf(15)
form38.Text37.Text = traf(16)
form38.Text38.Text = traf(17)
form38.Text39.Text = traf(18)
form38.Text40.Text = traf(19)
form38.Text41.Text = traf(20)
form38.Text42.Text = traf(21)
form38.Text43.Text = traf(22)
form38.Text44.Text = traf(23)
form38.Text45.Text = traf(24)
form38.Text46.Text = traf(25)
form38.Text47.Text = traf(26)
form38.Text48.Text = traf(27)
form38.Text49.Text = traf(28)
form38.Text50.Text = traf(29)
form38.Text51.Text = traf(30)
```

```
form31.Show
form1.Hide
End Sub
```

```
Sub PR_Click ()
form15.Text9.Text = "POR RUTAS"
form15.Show
form1.Hide
End Sub
```

```
Sub R_Click ()
```



```
Set dnaset = bd.CreateDynaset("carga_procesador_dia") 'ABRE LA TABLA UTILIZAR SUS DATOS
EN LA GRAFICA.
```

```
iniciov = form13.MaskedEdit1.Text 'PONEMOS LA FECHA DE TOMA DE DATOS.
```

```
Do While dnaset.EOF = False
    If iniciov = dnaset("inicio") Then
        var = dnaset("horap") 'ELABORACION DE LAS GRAFICAS
        form48.Graph10.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
        form48.Graph10.GraphData = dnaset("pload")
    End If
    dnaset.MoveNext
```

```
Loop
form48.Graph10.GraphTitle = "Carga de Procesador en Hora Pico"
form3.Text2.Text = var
dnaset.Close 'CIERRE DE LA BASE DE DATOS Y DYNASETS
bd.Close
form3.Show
```

PROCEDIMIENTO PARA TENER LA GRAFICA DE CARGA DEL PROCESADOR EN DIA PICO,  
TOMANDO

'EN CUENTA UNICAMENTE LAS MEDICIONES HECHAS EN CADA HORA DEL DIA.

```
'Dim db As database
```

```
'Dim dnaset As dynaset
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
Set dnaset = db.CreateDynaset("carga_procesador") 'ESTOS DATOS ESTAN ALMACENADOS EN LA
TABLA CARGA_PROCESADOR
```

```
'Dim var As Variant
```

```
iniciov = form13.MaskedEdit1.Text
```

```
Do While dnaset.EOF = False
```

```
    If iniciov = dnaset("inicio") Then
        var = dnaset("diap") 'ELABORACION DE LA GRAFICA
        form48.Graph9.ThisPoint = Val(Mid$(dnaset("hora"), 1, 2))
        form48.Graph9.GraphData = dnaset("pload")
    End If
    dnaset.MoveNext
```

```
Loop
form48.Graph9.GraphTitle = "Carga de Procesador en Dia Pico"
form2.Text2.Text = var
form2.Show
```

'ELABORACION DE LA GRAFICA DE LLAMADAS ENTRANTES POR SEGUNDO EN HORA  
PICO

'DATOS TOMADOS EN UNA HORA ESPECIFICA TODOS LOS DIAS DEL MES.

'ESTOS DATOS SE ENCUENTRAN EN ALMACENADOS EN LA TABLA  
CARGA\_PROCESADOR\_DIA

```
'Dim db As database
```

```
Dim dset As dynaset
```

```
'Dim var As Variant
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
Set dset = db.CreateDynaset("carga_procesador_dia")
```

```
iniciov = form13.MaskedEdit1.Text 'DETERMINAMOS LA FECHA DE TOMA DE DATOS.
```

```
Do While dset.EOF = False
```

```
    If iniciov = dset("inicio") Then
        var = dset("horap")
        form48.Graph13.ThisPoint = Val(Mid$(dset("dia"), 1, 2)) 'DAMOS VALORES A LOS
```

PARAMETROS

```
        form48.Graph13.GraphData = dset("fchcdi") 'PARA LA ELABORACION DEL
GRAFICO.
```

```

End If
dset.MoveNext
Loop
form48.Graph13.GraphTitle = "LLlamadas entrantes/seg hora pico"

```

```

Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("Sis_Perdida")
Set dat = db.CreateDynaset("Datos")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

```

```

Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            saliente(i) = naset("traff")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            entrante(j) = naset("traff")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop
tot = 0
For i = 0 To 35
    total(i) = saliente(i) + entrante(i)
    tot = tot + total(i)
Next i
Dim bet As Variant

```

```
Dim bet1 As Variant
```

```
bet = 0
```

```
bet = total(0) + total(3) + total(5) + total(12) + total(17) + total(20) + total(21) + total(22)
```

```
bet = bet + total(23) + total(24) + total(25)
```

```
bet1 = total(1) + total(2) + total(4) + total(7) + total(8) + total(9) + total(10) + total(11)
```

```
bet1 = bet + total(13) + total(14) + total(15) + total(16) + total(18) + total(19)
```

```
form48.Graph11.GraphData = bet1 * 100 / tot
```

```
form48.Graph11.GraphData = bet * 100 / tot
```

```
form48.Graph11.GraphData = total(15) * 100 / tot
```

```
form48.Graph11.GraphData = total(26) * 100 / tot
```

```
form48.Graph11.GraphData = total(27) * 100 / tot
```

```
form48.Graph11.GraphData = total(28) * 100 / tot
```

```
form48.Graph11.GraphData = total(29) * 100 / tot
```

```
form48.Graph11.GraphData = total(30) * 100 / tot
```

```
form48.Graph11.GraphData = total(31) * 100 / tot
```

```
form48.Graph11.LegendText = "Quito"
```

```
form48.Graph11.LegendText = "Area 2"
```

```
form48.Graph11.LegendText = "Tel. Movil"
```

```
form48.Graph11.LegendText = "Ambato Transito"
```

```
form48.Graph11.LegendText = "Cuenca Transito"
```

```
form48.Graph11.LegendText = "Guayaquil Transito"
```

```
form48.Graph11.LegendText = "Ibarra Transito"
```

```
form48.Graph11.LegendText = "Quito Transito 1"
```

```
form48.Graph11.LegendText = "Internacional"
```

```
form48.Graph11.GraphTitle = "Interés de Tráfico"
```

```
form48.Show
```

```
End Sub
```

```
Sub SALR_Click ()
```

```
End
```

```
End Sub
```

```
Sub SPC_Click ()
```

```
form12.Show
```

```
form1.Hide
```

```
End Sub
```

```
Dim externa As Variant
```

```
Sub Command1_Click ()  
form1.Show  
Unload form10  
form10.Hide  
End Sub
```

```
Sub Command3D1_Click ()  
form23.Show  
form10.Hide  
End Sub
```

```
Sub Command3D2_Click ()  
Dim db As database  
Dim dat As dynaset  
Dim sal, ent As Variant  
Dim naset As dynaset  
Set db = OpenDatabase("c:\vb\pruebas.mdb")  
Set dat = db.CreateDynaset("Datos")  
Set naset = db.CreateDynaset("Sis_Perdida")  
Dim i As Integer  
Dim r As Integer  
'Dim var As Variant  
ReDim VEC(35) As Variant  
ReDim abr(35) As Variant  
iniciov = form13.MaskedEdit1.Text  
i = 0  
Do While dat.EOF = False  
  
    VEC(i) = dat("central")  
    abr(i) = dat("abrev")  
  
    i = i + 1  
    dat.MoveNext  
Loop
```

```
form10.Hide
```

```
Unload form10
```

```
form43.Text1.Text = 10  
form43.Text29.Text = 11  
form43.Text3.Text = 12  
form43.Text4.Text = 13  
form43.Text5.Text = 14  
form43.Text6.Text = 15  
form43.Text7.Text = 16  
form43.Text8.Text = 17  
form43.Text9.Text = 18  
form43.Text10.Text = 19  
form43.Text11.Text = 20  
form43.Text12.Text = 21
```

```
form43.Text13.Text = 22
form43.Text14.Text = 23
form43.Text15.Text = 24
form43.Text16.Text = 25
```

```
form43.Text18.Text = VEC(9)
form43.Text19.Text = VEC(10)
form43.Text20.Text = VEC(11)
form43.Text21.Text = VEC(12)
form43.Text22.Text = VEC(13)
form43.Text23.Text = VEC(14)
form43.Text24.Text = VEC(16)
form43.Text25.Text = VEC(17)
form43.Text26.Text = VEC(18)
```

```
form43.Text27.Text = VEC(19)
form43.Text28.Text = VEC(20)
form43.Text29.Text = VEC(21)
form43.Text30.Text = VEC(22)
form43.Text31.Text = VEC(23)
form43.Text32.Text = VEC(24)
form43.Text33.Text = VEC(25)
```

```
form43.Text49.Text = abr(9)
form43.Text50.Text = abr(10)
form43.Text51.Text = abr(11)
form43.Text52.Text = abr(12)
form43.Text53.Text = abr(13)
form43.Text54.Text = abr(14)
form43.Text55.Text = abr(16)
form43.Text56.Text = abr(17)
form43.Text57.Text = abr(18)
```

```
form43.Text58.Text = abr(19)
form43.Text59.Text = abr(20)
form43.Text60.Text = abr(21)
form43.Text61.Text = abr(22)
form43.Text62.Text = abr(23)
form43.Text63.Text = abr(24)
form43.Text64.Text = abr(25)
```

```
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Set naset = db.CreateDynaset("Sis_Perdida")
Do While naset.EOF = False

    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            var(i) = naset("ndv")
            i = i + 1
        End If
    End If
End Do
```

```
naset.MoveNext
Loop
```

```
Set naset = db.CreateDynaset("sis_perdida")
i = 0
ReDim bloq(35) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            bloq(i) = naset("anblo")
            i = i + 1
        End If
    End If
    naset.MoveNext
```

```
Loop
```

```
Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            saliente(i) = naset("traff")
            i = i + 1
        End If
        If abr(j) + "I" = naset("ruta") Then
            entrante(j) = naset("traff")
            j = j + 1
        End If
    End If
    naset.MoveNext
```

```
Loop
```

```
Set naset = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            cong(i) = naset("ccong")
            i = i + 1
        End If
    End If
    naset.MoveNext
```

```
Loop
```

form43.Text150.Text = cong(9)  
form43.Text156.Text = cong(10)  
form43.Text157.Text = cong(11)  
form43.Text158.Text = cong(12)  
form43.Text159.Text = cong(13)  
form43.Text160.Text = cong(14)  
form43.Text139.Text = cong(15)  
form43.Text128.Text = cong(16)  
form43.Text129.Text = cong(17)

form43.Text130.Text = cong(18)  
form43.Text131.Text = cong(19)  
form43.Text132.Text = cong(20)  
form43.Text133.Text = cong(21)  
form43.Text134.Text = cong(22)  
form43.Text135.Text = cong(23)  
form43.Text136.Text = cong(24)

form43.Text163.Text = saliente(9)  
form43.Text165.Text = saliente(10)  
form43.Text117.Text = saliente(11)  
form43.Text118.Text = saliente(12)  
form43.Text119.Text = saliente(13)  
form43.Text120.Text = saliente(14)  
form43.Text121.Text = saliente(15)  
form43.Text122.Text = saliente(16)  
form43.Text123.Text = saliente(17)  
form43.Text124.Text = saliente(18)

form43.Text125.Text = saliente(19)  
form43.Text126.Text = saliente(20)  
form43.Text127.Text = saliente(21)  
form43.Text99.Text = saliente(22)  
form43.Text98.Text = saliente(23)

form43.Text161.Text = bloq(9)  
form43.Text148.Text = bloq(10)  
form43.Text149.Text = bloq(11)  
form43.Text100.Text = bloq(12)  
form43.Text84.Text = bloq(13)  
form43.Text80.Text = bloq(14)  
form43.Text81.Text = bloq(15)

```
form43.Text82.Text = bloq(16)
form43.Text83.Text = bloq(17)
```

```
form43.Text91.Text = bloq(18)
form43.Text108.Text = bloq(19)
form43.Text109.Text = bloq(20)
form43.Text107.Text = bloq(21)
form43.Text110.Text = bloq(22)
form43.Text111.Text = bloq(23)
form43.Text112.Text = bloq(24)
'para mostrar datos de traff
```

```
form43.Text162.Text = entrante(9)
form43.Text166.Text = entrante(10)
form43.Text167.Text = entrante(11)
form43.Text96.Text = entrante(12)
form43.Text95.Text = entrante(13)
form43.Text94.Text = entrante(14)
form43.Text93.Text = entrante(15)
form43.Text77.Text = entrante(16)
form43.Text76.Text = entrante(17)
```

```
form43.Text78.Text = entrante(18)
form43.Text79.Text = entrante(19)
form43.Text101.Text = entrante(20)
form43.Text102.Text = entrante(21)
form43.Text103.Text = entrante(22)
form43.Text104.Text = entrante(23)
form43.Text105.Text = entrante(24)
```

```
'para mostrar ccong
```

```
form43.Text71.Text = cong(9)
form43.Text72.Text = cong(10)
form43.Text73.Text = cong(11)
form43.Text74.Text = cong(12)
form43.Text37.Text = cong(13)
form43.Text38.Text = cong(14)
form43.Text39.Text = cong(15)
form43.Text40.Text = cong(16)
form43.Text36.Text = cong(17)
```

```
form43.Text41.Text = cong(18)
form43.Text42.Text = cong(19)
form43.Text43.Text = cong(20)
form43.Text44.Text = cong(21)
form43.Text45.Text = cong(22)
form43.Text46.Text = cong(23)
form43.Text47.Text = cong(24)
```

```
form43.Show
```

```
End Sub
```



```

Dim numer As Variant

Sub Command1_Click ()
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_perdida")
iniciov = form13.MaskedEdit1.Text
Dim var As Variant
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            dbnaset.Edit
            dbnaset("ccong") = text5
            dbnaset("anblo") = text6
            dbnaset("mhtime") = text7
            dbnaset("nbansw") = text8
            dbnaset.Update
        End If
    End If
    dbnaset.MoveNext
Loop

    MsgBox ("datos grabados")

    form18.Show
End Sub

Sub Command2_Click ()
form1.Show
Unload form11
form11.Hide

End Sub

Sub Command3_Click ()
form1.Show
form11.Hide
text1 = ""
text2 = ""
text3 = ""
text4 = ""
text5 = ""
text6 = ""
text7 = ""
text8 = ""

End Sub

Sub Text1_LostFocus ()
text1.Text = UCase$(text1.Text)
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_perdida")
Dim ban As Variant
Dim bet As Variant
Dim con As Variant
iniciov = form13.MaskedEdit1.Text

```

```

ban = 1
con = 0
bet = 0
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            ban = 0
            If dbnaset("ccong") = 0 Then
                text2.Text = dbnaset("traff")
                text3.Text = dbnaset("nbids")
                text4.Text = dbnaset("ndv")
                text5.SetFocus
            Else
                bet = 1
            End If
        End If
    End If
    dbnaset.MoveNext
Loop

If ban = 1 Then
    text1.Text = ""
    MsgBox ("Valores incompletos ")

End If

If bet = 1 Then
    text1.Text = ""
    MsgBox ("ESTOS DATOS YA ESTAN REGISTRADOS")

End If

End Sub

Sub Text2_KeyPress (keyascii As Integer)
Static puntodecimalusado As Integer
Dim ayuda As Variant
If keyascii > Asc("a") And keyascii < Asc("z") Then
    Beep
    keyascii = 0
    text2.Text = ""
Else
    Select Case keyascii
    Case Asc("0") To Asc("9")
        ayuda = ayuda + Chr$(keyascii)
    Case Asc(".")
        If puntodecimalusado Then
            keyascii = 0
            Beep
        Else
            puntodecimalusado = True
            ayuda = ayuda + Chr$(keyascii)
        End If
    End Select
End Sub

```

```
        text2.Text = ayuda
        Case Asc(",")
            ' coma no afecta
        Case Else
            keyascii = 0
            Beep
    End Select

    End If

End Sub

Sub Text3_KeyPress (keyascii As Integer)
    If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
    Asc("Z") Then
        Beep
        keyascii = 0
    End If
End Sub

Sub Text4_KeyPress (keyascii As Integer)
    If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
    Asc("Z") Then
        Beep
        keyascii = 0
        text4.Text = ""
    End If
End Sub

Sub Text5_KeyPress (keyascii As Integer)
    If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
    Asc("Z") Then
        Beep
        keyascii = 0
        text5.Text = ""
    End If
End Sub

Sub Text6_KeyPress (keyascii As Integer)
    If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
    Asc("Z") Then
        Beep
        keyascii = 0
        text6.Text = ""
    End If
End Sub

Sub Text7_KeyPress (keyascii As Integer)
    If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
    Asc("Z") Then
        Beep
        keyascii = 0
        text7.Text = ""
    End If
End Sub
```

End Sub

Sub Text8\_Change ()

If text8.Text <> "" Then

    command1.Enabled = True

End If

End Sub

Sub Text8\_KeyPress (keyascii As Integer)

If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii < Asc("Z") Then

    Beep

    keyascii = 0

    text8.Text = ""

End If

End Sub

```
Sub Command1_Click ()
```

```
Dim db As database
```

```
Dim naset As dynaset
```

```
Dim ncola As dynaset
```

```
Dim nperdida As dynaset
```

```
Dim ncalidad As dynaset
```

```
Dim hora As dynaset
```

```
Dim distrafico As dynaset
```

```
Dim otrafico As dynaset
```

```
Dim mtp As dynaset
```

```
Dim dia As dynaset
```

```
Dim saliente As dynaset
```

```
Set db = OpenDatabase("c:\vb\pruebas.mdb")
```

```
'Set naset = db.CreateDynaset("fecha")
```

```
Set ncola = db.CreateDynaset("sis_cola")
```

```
Set nperdida = db.CreateDynaset("sis_perdida")
```

```
Set ncalidad = db.CreateDynaset("calidad_servicio")
```

```
Set hora = db.CreateDynaset("carga_procesador")
```

```
Set dia = db.CreateDynaset("carga_procesador_dia")
```

```
Set otrafico = db.CreateDynaset("observacion_trafico")
```

```
Set mtp = db.CreateDynaset("mtp_trafico")
```

```
Set distrafico = db.CreateDynaset("disp_trafico")
```

```
Set saliente = db.CreateDynaset("trafico_saliente")
```

```
'ncola.AddNew 'actualiza datos en la tabla sis_cola
```

```
'ncola("inicio") = MaskedEdit1.Text
```

```
'ncola("final") = MaskedEdit2.Text
```

```
'ncola("diap") = MaskedEdit3.Text
```

```
'ncola("horap") = MaskedEdit4.Text
```

```
'ncola("ruta") = "n"
```

```
'ncola.Update
```

```
'ncola.Close
```

```
'nperdida.AddNew 'actualiza datos en la tabla sis_perdida
```

```
'nperdida("inicio") = MaskedEdit1.Text
```

```
'nperdida("final") = MaskedEdit2.Text
```

```
'nperdida("diap") = MaskedEdit3.Text
```

```
'nperdida("horap") = MaskedEdit4.Text
```

```
'nperdida("ruta") = "n"
```

```
'nperdida.Update
```

```
'nperdida.Close
```

```
'distrafico.AddNew 'actualiza datos en la tabla disp_trafico
```

```
'distrafico("inicio") = MaskedEdit1.Text
```

```
'distrafico("final") = MaskedEdit2.Text
```

```
'distrafico("diap") = MaskedEdit3.Text
```

```
'distrafico("horap") = MaskedEdit4.Text
```

```
'distrafico("tipo") = "n"
```

```
'distrafico.Update
```

```
'distrafico.Close
```

```
MsgBox ("Datos grabados")
iniciov = MaskedEdit1.Text
form16.Show
form13.Hide
End Sub

Sub Command2_Click ()
MaskedEdit1.Text = "01-05-1995"
MaskedEdit2.Text = "31-05-1995"
MaskedEdit3.Text = "25-05-1995"
MaskedEdit4.Text = "09:30 am a 10:30 am"

form13.Hide
Load form1
form1.Show

End Sub

Sub MaskedEdit1_LostFocus ()
Dim db As database
Dim dbnaset As dynaset
Dim ban As Variant

Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_perdida")
ban = 0

Do While dbnaset.EOF = False
    If MaskedEdit1.Text = dbnaset("inicio") Then
        ban = 1
    End If
    dbnaset.MoveNext
Loop

If ban = 1 Then
    MsgBox ("Este periodo ya tiene Valores")
    MaskedEdit1.SetFocus
End If

End Sub

Sub MaskedEdit4_LostFocus ()
command1.Enabled = True

End Sub

Sub Text1_KeyPress (keyascii As Integer)
If (keyascii > Asc("a") And keyascii < Asc("z")) Or (keyascii > Asc("A") And keyascii < Asc("Z")) Then
    Beep
    keyascii = 0
    'text1.Text = ""
End If

End Sub

Sub Text1_LostFocus ()
Dim db As database
```

```
Dim naset As dynaset
Dim ncola As dynaset
Dim nperdida As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("fecha")
Set ncola = db.CreateDynaset("sisCola")
Set nperdida = db.CreateDynaset("sis_perdida")
Dim x1 As Variant
Dim x2 As Variant
Dim x3 As Variant
Dim x4 As Variant
Dim x5 As Variant
Dim x6 As Variant
Dim variable As Variant
variable = Date$
x1 = Mid$(variable, 1, 2)
x2 = Mid$(variable, 4, 2)
x5 = Mid$(variable, 7, 10)
MsgBox (x5)
Do While ncola.EOF = False
    If ncola("inicio") <> "" Then
        x3 = Mid$(ncola("inicio"), 1, 2)
        x4 = Mid$(ncola("inicio"), 4, 2)
        x6 = Mid$(ncola("inicio"), 7, 10)
        If Val(x5) = Val(x6) Then
            If Val(x1) = (x3) Then
                MsgBox ("En este periodo se estan realizando la toma de datos")
                Beep
            End If
        End If
    End If
    ncola.MoveNext
Loop
End Sub
```

```

Sub Command1_Click ()
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("mtp_ trafico")
iniciov = form13.MaskedEdit1.Text
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then

            dbnaset("ruta") = text1.Text
            If option3d1.Value = -1 Then
                dbnaset.Edit
                dbnaset("originado") = "si"
                Else ' nivel1
                    dbnaset("originado") = "no"
                    If option3d2.Value = -1 Then
                        dbnaset("interno") = "si"
                    Else nivel2
                        dbnaset("interno") = "no"

                    If option3d3.Value = -1 Then
                        dbnaset("terminal") = "si"
                    Else nivel 3
                        dbnaset("terminal") = "no"
                        If option3d4.Value = -1 Then
                            dbnaset("transito") = "si"
                        Else
                            dbnaset("transito") = "no"

                        End If
                    End If

                End If
            End If

            'dbnaset("erl") = InputBox("ingrese los erlangs")
            'dbnaset("DIA") = InputBox("ingrese el dia")
            dbnaset.Edit
            dbnaset("unsuc") = Text4.Text

            dbnaset("ndvo") = Text5.Text
            dbnaset("ndub") = text6.Text
            dbnaset("nbloo") = text7.Text
            dbnaset("nblob") = text8.Text
            dbnaset.Update
        End If
        dbnaset.MoveNext
    Loop
    MsgBox ("Datos grabados")

End Sub

Sub Command2_Click ()
FORM1.Show
FORM14.Hide
End Sub

Sub Text1_KeyPress (keyascii As Integer)

```



```

If keyascii < Asc("a") Or keyascii > Asc("z") Then
    Beep
    keyascii = 0
    text1.Text = ""
End If
End Sub

Sub Text1_LostFocus ()
text1.Text = UCase$(text1.Text)
Dim caracter As Variant
caracter = Mid$(text1.Text, 1, 3)
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("mtp_trafico")
iniciov = form13.MaskedEdit1.Text
ban = 1
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            ban = 0
            Text2.Text = dbnaset("traff")
            Text3.Text = dbnaset("nbids")
            Text4.SetFocus
        End If
    End If
    dbnaset.MoveNext
Loop
    If ban = 1 Then
        text1.Text = ""
        MsgBox ("No ingresados los Valores generales")
        text1.SetFocus
    End If

End Sub

Sub Text2_KeyPress (keyascii As Integer)
Static puntodecimalusado As Integer
Dim ayuda As Variant
If keyascii > Asc("a") And keyascii < Asc("z") Then
    Beep
    keyascii = 0
    Text2.Text = ""
Else
    Select Case keyascii
    Case Asc("0") To Asc("9")
        ayuda = ayuda + Chr$(keyascii)
    Case Asc(".")
        If puntodecimalusado Then
            keyascii = 0
            Beep
        Else
            puntodecimalusado = True
            ayuda = ayuda + Chr$(keyascii)
        End If
    End Select
End If

```

```
        Text2.Text = ayuda
    Case Asc(",")
        ' coma no afecta
    Case Else
        keyascii = 0
        Beep
End Select
End If
End Sub

Sub Text3_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    Text3.Text = ""
End If

End Sub

Sub Text4_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    Text4.Text = ""
End If

End Sub

Sub Text5_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    Text5.Text = ""
End If

End Sub

Sub Text6_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    text6.Text = ""
End If

End Sub

Sub Text7_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then
    Beep
    keyascii = 0
    text7.Text = ""
End If

End Sub

Sub Text8_Change ()
If text8.Text <> "" Then
    command1.Enabled = True
End If
```

End Sub

Sub Text8\_KeyPress (keyascii As Integer)

If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Then

    Beep

    keyascii = 0

    text8.Text = ""

End If

End Sub

```

Sub Command1_Click ()
Dim db As database
Dim dbnaset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("disp_trafico")
iniciov = form13.MaskedEdit1.Text
p = 1
Do While dbnaset.EOF = False
    If dbnaset("inicio") = iniciov And dbnaset("ruta") = text1.Text Then
        dbnaset.Edit
        dbnaset("tipo") = text9.Text
        dbnaset("ncalls") = text2.Text
        dbnaset("noutgs") = text3.Text
        dbnaset("nres") = text4.Text
        dbnaset("ahtime") = text5.Text
        dbnaset("actime") = text8.Text
        dbnaset("nbansn") = text6.Text
        dbnaset.Update
    End If
    dbnaset.MoveNext
Loop

```

```

MsgBox ("Datos Grabados")
FORM24.Show
form15.Enabled = False
End Sub

```

```

Sub Command2_Click ()
Form1.Show
Unload form15

```

```
End Sub
```

```

Sub Text1_KeyPress (KeyAscii As Integer)
If KeyAscii > Asc("0") And KeyAscii < Asc("9") Then
    Beep
    KeyAscii = 0
    text1.Text = ""
End If

```

```
End Sub
```

```

Sub Text1_LostFocus ()
text1.Text = UCase$(text1.Text)
Dim db As database
Dim dbnaset As dynaset
Dim naset As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("disp_trafico")
Set naset = db.CreateDynaset("sis_perdida")
Dim ban As Variant
Dim con As Variant
iniciov = form13.MaskedEdit1.Text
ban = 1
con = 0
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then

```

```

        If iniciov = dbnaset("inicio") Then
            If dbnaset("ncalls") = 0 Then
                ban = 0
                If dbnaset("traff") <> 0 Then
                    text7.Text = dbnaset("traff")
                    text2.SetFocus
                End If
            Else
                con = 1
            End If
        End If
    End If
    dbnaset.MoveNext
Loop

Set dbnaset = db.CreateDynaset("disp_trafico")
ban = 1
Do While dbnaset.EOF = False
    If text1.Text = dbnaset("ruta") Then
        If iniciov = dbnaset("inicio") Then
            ban = 0
        End If
    End If
    dbnaset.MoveNext
Loop

If ban = 1 Then
    MsgBox ("Datos incompletos")
End If

If con = 1 Then
    MsgBox ("Datos Registrados")
End If

End Sub

Sub Text2_KeyPress (KeyAscii As Integer)
    If KeyAscii >= Asc("a") And KeyAscii <= Asc("z") Or KeyAscii = 255 Or KeyAscii >= Asc("A") And
    KeyAscii <= Asc("Z") Then

        Beep
        KeyAscii = 0

    End If

End Sub

Sub Text2_LostFocus ()
    If Len(text2.Text) = 8 Then
        text3.SetFocus
    End If
End Sub

```

End If

End Sub

Sub Text3\_KeyPress (KeyAscii As Integer)

If KeyAscii > Asc("a") And KeyAscii < Asc("z") Or KeyAscii = 255 Or KeyAscii > Asc("A") And KeyAscii < Asc("Z") Then

Beep  
KeyAscii = 0  
text3.Text = ""

End If

End Sub

Sub Text4\_KeyPress (KeyAscii As Integer)

If KeyAscii > Asc("a") And KeyAscii < Asc("z") Or KeyAscii = 255 Or KeyAscii > Asc("A") And KeyAscii < Asc("Z") Then

Beep  
KeyAscii = 0  
text4.Text = ""

End If

End Sub

Sub Text5\_KeyPress (KeyAscii As Integer)

If KeyAscii > Asc("a") And KeyAscii < Asc("z") Or KeyAscii = 255 Or KeyAscii > Asc("A") And KeyAscii < Asc("Z") Then

Beep  
KeyAscii = 0  
text5.Text = ""

End If

End Sub

Sub Text6\_KeyPress (KeyAscii As Integer)

If KeyAscii > Asc("a") And KeyAscii < Asc("z") Or KeyAscii = 255 Or KeyAscii > Asc("A") And KeyAscii < Asc("Z") Then

Beep  
KeyAscii = 0  
text6.Text = ""

End If

End Sub

Sub Text6\_LostFocus ()

If text6.Text <> "" Then  
command1.Enabled = True  
End If

End Sub

Sub Text8\_KeyPress (KeyAscii As Integer)

If KeyAscii > Asc("a") And KeyAscii < Asc("z") Or KeyAscii = 255 Or KeyAscii > Asc("A") And KeyAscii < Asc("Z") Then

Beep  
KeyAscii = 0

End If

```

Sub Command1_Click ()
Dim db As database
Dim dbnaset As dynaset
Dim ncolas As dynaset
Dim ntipo As dynaset
Dim ndispersion As dynaset
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("Sis_perdida")
Set ncolas = db.CreateDynaset("Sis_cola")
Set ntipo = db.CreateDynaset("ntp_trafico")
Set ndispersion = db.CreateDynaset("disp_trafico")
'Set saliente = db.CreateDynaset("trafico_saliente")

```

'actualiza datos en la tabla sis\_perdida \*\*\*\*\*

```

dbnaset.AddNew
dbnaset("ruta") = Text1.Text + "I"
dbnaset("traff") = text2.Text
dbnaset("nbids") = text3.Text
dbnaset("ndv") = text4.Text
dbnaset("ccong") = 0
dbnaset("anblo") = 0
dbnaset("mhitime") = 0
dbnaset("nbansw") = 0
dbnaset("inicio") = form13.MaskedEdit1.Text
dbnaset("final") = form13.MaskedEdit2.Text
dbnaset("diap") = form13.MaskedEdit3.Text
dbnaset("horap") = form13.MaskedEdit4.Text
dbnaset.Update

```

'para rutas salientes

```

dbnaset.AddNew
dbnaset("ruta") = Text1 + "O"
dbnaset("traff") = text5.Text
dbnaset("nbids") = text6.Text
dbnaset("ndv") = text7.Text
dbnaset("ccong") = 0
dbnaset("anblo") = 0
dbnaset("mhitime") = 0
dbnaset("nbansw") = 0
dbnaset("inicio") = form13.MaskedEdit1.Text
dbnaset("final") = form13.MaskedEdit2.Text
dbnaset("diap") = form13.MaskedEdit3.Text
dbnaset("horap") = form13.MaskedEdit4.Text
dbnaset.Update

```

'actualizacion de datos en la tabla sis\_cola \*\*\*\*\*

```

ncolas.AddNew
ncolas("ruta") = Text1 + "O"

```

```

ncolas("traff") = text5.Text
ncolas("nbids") = text6.Text
ncolas("ndv") = text7.Text
ncolas("nbidsq") = 0
ncolas("nsaq") = 0
ncolas("anblo") = 0
ncolas("nhtime") = 0
ncolas("mqtime") = 0
ncolas("inicio") = form13.MaskedEdit1.Text
ncolas("final") = form13.MaskedEdit2.Text
ncolas("diap") = form13.MaskedEdit3.Text
ncolas("horap") = form13.MaskedEdit4.Text
ncolas.Update

```

```

ncolas.AddNew
ncolas("ruta") = Text1 + "I"
ncolas("traff") = text2.Text
ncolas("nbids") = text3.Text
ncolas("ndv") = text4.Text
ncolas("nbidsq") = 0
ncolas("nsaq") = 0
ncolas("anblo") = 0
ncolas("nhtime") = 0
ncolas("mqtime") = 0
ncolas("inicio") = form13.MaskedEdit1.Text
ncolas("final") = form13.MaskedEdit2.Text
ncolas("diap") = form13.MaskedEdit3.Text
ncolas("horap") = form13.MaskedEdit4.Text
ncolas.Update

```

'actualizacion de datos en la tabla tipo de trafico

```

r = 0
p = 1

```

```

ntipo.AddNew

ntipo("traff") = text2.Text
ntipo("nbids") = text3.Text
ntipo("unsuc") = 0
ntipo("ndvo") = 0
ntipo("ndvb") = 0
ntipo("nblo") = 0
ntipo("nblov") = 0
ntipo("ndvb") = 0
ntipo("inicio") = form13.MaskedEdit1.Text
ntipo("final") = form13.MaskedEdit2.Text
ntipo("diap") = form13.MaskedEdit3.Text
ntipo("horap") = form13.MaskedEdit4.Text
ntipo.Update

```

'actualizacion de datos en la tabla dispersion

```

r = 0

```



p = 1

```

ndispersion.AddNew
ndispersion("ruta") = Text1 + "O"
ndispersion("traff") = text5.Text
ndispersion("ncalls") = 0
ndispersion("noutgs") = 0
ndispersion("nres") = 0
ndispersion("nhtime") = 0
ndispersion("actime") = 0
ndispersion("nbansn") = 0

```

```

ndispersion("inicio") = form13.MaskedEdit1.Text
ndispersion("final") = form13.MaskedEdit2.Text
ndispersion("diap") = form13.MaskedEdit3.Text
ndispersion("horap") = form13.MaskedEdit4.Text
ndispersion.Update

```

```

ndispersion.AddNew
ndispersion("ruta") = Text1 + "I"
ndispersion("traff") = text2.Text
ndispersion("ncalls") = 0
ndispersion("noutgs") = 0
ndispersion("nres") = 0
ndispersion("nhtime") = 0
ndispersion("actime") = 0
ndispersion("nbansn") = 0
ndispersion("inicio") = form13.MaskedEdit1.Text
ndispersion("final") = form13.MaskedEdit2.Text
ndispersion("diap") = form13.MaskedEdit3.Text
ndispersion("horap") = form13.MaskedEdit4.Text
ndispersion.Update

```

```

MsgBox ("datos grabados")
form17.Show

```

End Sub

```

Sub Command2_Click ()
form1.Show
form16.Hide
form13.Hide

```

End Sub

```

Sub Text1_LostFocus ()
Text1.Text = UCase$(Text1.Text)
Dim db As database
Dim dbnaset As dynaset
Dim i As Variant
Dim i1 As Variant

```

```
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii < Asc("Z") Then
```

```
    Beep
    keyascii = 0
    text6.Text = ""
```

```
End If
```

```
End Sub
```

```
Sub Text7_KeyPress (keyascii As Integer)
```

```
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii < Asc("Z") Then
```

```
    Beep
    keyascii = 0
    text7.Text = ""
```

```
End If
```

```
End Sub
```

```
Sub Text7_LostFocus ()
```

```
command1.Enabled = True
```

```
End Sub
```

```
Sub Command1_Click ()  
Unload form16  
Load form16  
form16.Show  
Unload form17
```

```
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
form13.Hide  
Unload form16  
Unload form17  
End Sub
```

```
Sub Command1_Click ()  
Unload form11  
form11.Show  
form18.Hide  
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
Unload form18  
Unload form11  
form18.Hide  
form11.Hide  
  
End Sub
```

```
Sub Command1_Click ()
form19.Hide
form12.Command1.Enabled = False
form12.Text1.Text = ""
form12.Text2.Text = ""
form12.Text3.Text = ""
form12.Text4.Text = ""
form12.Text5.Text = ""
form12.Text6.Text = ""
form12.Text7.Text = ""
form12.Text8.Text = ""
form12.Text9.Text = ""
form12.Text1.SetFocus
End Sub
```

```
Sub Command2_Click ()
form1.Show
form12.Hide
form19.Hide

End Sub
```

```
Sub Command1_Click ()
Dim db As database
Dim dbnaset As dynaset
Dim naset As dynaset
Dim ban As Variant
Dim r, p As Integer
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
Set dbnaset = db.CreateDynaset("mtp_ trafico")
Set naset = db.CreateDynaset("trafico_saliente")
iniciov = form13.MaskedEdit1.Text
r = 0
p = 0

    dbnaset.AddNew

        If option3d1.Value = -1 Then
            dbnaset("originado") = 1
            dbnaset("interno") = 0
            dbnaset("terminal") = 0
            dbnaset("transito") = 0
            dbnaset("entrante") = 0
            dbnaset("saliente") = 0

        End If

        If option3d2.Value = -1 Then
            dbnaset("interno") = 1
            dbnaset("originado") = 0
            dbnaset("terminal") = 0
            dbnaset("transito") = 0
            dbnaset("entrante") = 0
            dbnaset("saliente") = 0

        End If

        If option3d3.Value = -1 Then
            dbnaset("terminal") = 1
            dbnaset("interno") = 0
            dbnaset("originado") = 0
            dbnaset("transito") = 0
            dbnaset("entrante") = 0
            dbnaset("saliente") = 0

        End If

        If option3d4.Value = -1 Then
            dbnaset("transito") = 1
            dbnaset("interno") = 0
            dbnaset("terminal") = 0
            dbnaset("originado") = 0
            dbnaset("entrante") = 0
            dbnaset("saliente") = 0

        End If
```

```
If option3d5.Value = -1 Then
    dbnaset("transito") = 0
    dbnaset("interno") = 0
    dbnaset("terminal") = 0
    dbnaset("originado") = 0
    dbnaset("entrante") = 1
    dbnaset("saliente") = 0
```

```
End If
```

```
If option3d6.Value = -1 Then
    dbnaset("transito") = 0
    dbnaset("interno") = 0
    dbnaset("terminal") = 0
    dbnaset("originado") = 0
    dbnaset("entrante") = 0
    dbnaset("saliente") = 1
```

```
End If
```

```
dbnaset("dia") = MaskedEdit1.Text
dbnaset("traff") = text3.Text
dbnaset("nbids") = Text4.Text
dbnaset("unsuc") = text5.Text
dbnaset("ndvo") = text6.Text
dbnaset("ndvb") = text7.Text
dbnaset("nbloo") = TEXT8.Text
dbnaset("nblov") = text9.Text
dbnaset("inicio") = form13.MaskedEdit1.Text
dbnaset("final") = form13.MaskedEdit2.Text
dbnaset("diap") = form13.MaskedEdit3.Text
dbnaset("horap") = form13.MaskedEdit4.Text
dbnaset.Update
dbnaset.Close
MsgBox ("Datos Grabados")
```

```
If option3d6.Value = -1 Then
```

```
    naset.AddNew
    naset("dia") = MaskedEdit1.Text
    naset("traff") = text3.Text
    naset("nbids") = Text4.Text
    naset("unsuc") = text5.Text
    naset("ndvo") = text6.Text
    naset("ndvb") = text7.Text
    naset("nbloo") = TEXT8.Text
    naset("nblov") = text9.Text
    naset("inicio") = form13.MaskedEdit1.Text
    naset("final") = form13.MaskedEdit2.Text
    naset("diap") = form13.MaskedEdit3.Text
    naset("horap") = form13.MaskedEdit4.Text
    naset.Update
    naset.Close
```

```
MsgBox ("trafico Saliente")

End If

form21.Show
End Sub

Sub Command2_Click ()
Text1.Text = ""
text3.Text = ""
Text4.Text = ""
text5.Text = ""
text6.Text = ""
text7.Text = ""
TEXT8.Text = ""
text9.Text = ""
form1.Show
form20.Hide

End Sub

Sub MaskedEdit1_LostFocus ()
text3.SetFocus
End Sub

Sub MaskedEdit1_ValidationError (InvalidText As String, StartPosition As Integer)
MsgBox ("Ingrese la fecha (Formato Indicado)")
MaskedEdit1.SetFocus
End Sub

Sub Text1_LostFocus ()

End Sub

Sub Text3_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
    Beep
    keyascii = 0
    text3.Text = ""
End If

End Sub

Sub Text4_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
    Beep
    keyascii = 0
    Text4.Text = ""
End If

End Sub

Sub Text5_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
```



```
                Beep
                keyascii = 0
                text5.Text = ""
End If

End Sub

Sub Text6_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
                Beep
                keyascii = 0
                text6.Text = ""
End If

End Sub

Sub Text7_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
                Beep
                keyascii = 0
                text7.Text = ""
End If

End Sub

Sub Text8_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
                Beep
                keyascii = 0
                TEXT8.Text = ""
End If

End Sub

Sub Text9_KeyPress (keyascii As Integer)
If keyascii > Asc("a") And keyascii < Asc("z") Or keyascii = 255 Or keyascii > Asc("A") And keyascii <
Asc("Z") Then
                Beep
                keyascii = 0
                text9.Text = ""
End If

End Sub

Sub Text9_LostFocus ()
command1.Enabled = True
End Sub
```

---

```
Sub Command1_Click ()  
form21.Hide  
Unload form20  
Unload form21  
form20.Show  
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
form21.Hide  
form20.Hide  
End Sub
```

```

Sub Command3_Click ()
form23.Hide
Unload form23
Unload form10
'Load form45
Unload form31
Unload form38

```

```

'Unload form1
Unload form2
Unload form3
Unload form4
Unload form5
Unload form6
Unload form7
Unload form8
Unload form9
Unload form11
Unload form12
'Unload form13

```

```

Unload form15
Unload form16
Unload form17
Unload form18

```

INSTRUCCIONES NECESARIAS PARA LA ELABORACION DE LOS  
'CUADROS ESTADISTICOS, EN EL QUE SE DETALLARAN, DATOS COMO:  
'NOMBRE DE LA CENTRAL, CIRCUITOS BIDIREC. BLOQUEDOS EN HP,  
'CARGA DE PROCESADOR EN HORA PICO ETC.

```

Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
'Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

```

```

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

```

```

    i = i + 1
    dat.MoveNext

```

```

Loop
form10.Text90.Text = 1
form10.Text82.Text = 2
form10.Text83.Text = 3
form10.Text84.Text = 4
form10.Text85.Text = 5

```

```
form10.Text86.Text = 6
form10.Text87.Text = 7
form10.Text88.Text = 8
form10.Text89.Text = 9
```

```
form10.Text1.Text = VEC(0)
form10.Text2.Text = VEC(1)
form10.Text3.Text = VEC(2)
form10.Text4.Text = VEC(3)
form10.Text5.Text = VEC(4)
form10.Text6.Text = VEC(5)
form10.Text7.Text = VEC(6)
form10.Text8.Text = VEC(7)
form10.Text11.Text = VEC(8)
```

```
form10.Text12.Text = abr(0)
form10.Text13.Text = abr(1)
form10.Text14.Text = abr(2)
form10.Text15.Text = abr(3)
form10.Text16.Text = abr(4)
form10.Text17.Text = abr(5)
form10.Text18.Text = abr(6)
form10.Text10.Text = abr(7)
form10.Text9.Text = abr(8)
'DATOS PARA LA FORMA 43
```

```
i = 0
'BLOQUE PARA HALLAR EL VALOR DE CIRCUTOS BIDIRECCIONALES POR RUTA
'LOS QUE ESTAN EN LA TABLA SIS_PERDIDA
Set dat = db.CreateDynaset("sis_perdida")
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Do While dat.EOF = False

    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            var(i) = dat("ndv")
            i = i + 1
        End If
    End If
    dat.MoveNext
Loop

ReDim bloq(35) As Variant
Set dat = db.CreateDynaset("sis_perdida")
i = 0
Do While dat.EOF = False
    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            bloq(i) = dat("anblo")
            i = i + 1
        End If
    End If
End While
```

```
End If
dat.MoveNext
```

```
Loop
```

```
form10.Text27.Text = var(0)
form10.Text19.Text = var(1)
form10.Text20.Text = var(2)
form10.Text21.Text = var(3)
form10.Text22.Text = var(4)
form10.Text23.Text = var(5)
form10.Text24.Text = var(6)
form10.Text25.Text = var(7)
form10.Text26.Text = var(8)
FINAL DEL PRIMER BLOQUE
SEGUNDO BLOQUE
form10.Text37.Text = bloq(0)
form10.Text38.Text = bloq(1)
form10.Text39.Text = bloq(2)
form10.Text44.Text = bloq(3)
form10.Text45.Text = bloq(4)
form10.Text40.Text = bloq(5)
form10.Text46.Text = bloq(6)
form10.Text47.Text = bloq(7)
form10.Text48.Text = bloq(8)
```

```
'instrucciones para hallar el trafico los valores del trafico saliente
y entrante y determinar la suma
```

```
Set dat = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While dat.EOF = False
  If iniciov = dat("inicio") Then

    If abr(i) + "O" = dat("ruta") Then
      saliente(i) = dat("traff")
      i = i + 1
    End If

    If abr(j) + "I" = dat("ruta") Then
      entrante(j) = dat("traff")
      j = j + 1
    End If
  End If
  dat.MoveNext
Loop
```

```
'instrucciones para hallar el total
For i = 0 To 35
  total(i) = saliente(i) + entrante(i)
Next i
```

```
'bloque para visualizar los valores de trafico entrante y saliente
```

```
form10.Text42.Text = entrante(0)
form10.Text43.Text = entrante(1)
```

```

form10.Text49.Text = entrante(2)
form10.Text50.Text = entrante(3)
form10.Text51.Text = entrante(4)
form10.Text52.Text = entrante(5)
form10.Text53.Text = entrante(6)
form10.Text54.Text = entrante(7)
form10.Text55.Text = entrante(8)

```

```

form10.Text63.Text = saliente(0)
form10.Text62.Text = saliente(1)
form10.Text61.Text = saliente(2)
form10.Text60.Text = saliente(3)
form10.Text59.Text = saliente(4)
form10.Text58.Text = saliente(5)
form10.Text57.Text = saliente(6)
form10.Text56.Text = saliente(7)
form10.Text41.Text = saliente(8)

```

```

form10.Text65.Text = total(0)
form10.Text64.Text = total(1)
form10.Text74.Text = total(2)
form10.Text75.Text = total(3)
form10.Text76.Text = total(4)
form10.Text77.Text = total(5)
form10.Text78.Text = total(6)
form10.Text79.Text = total(7)
form10.Text80.Text = total(8)

```

halla el valor de porcentaje de congestión de tráfico saliente

```

Set dat = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While dat.EOF = False
    If inicio = dat("inicio") Then
        If abr(i) + "0" = dat("ruta") Then
            cong(i) = dat("ccong")
            i = i + 1
        End If
    End If
    dat.MoveNext

```

Loop

```

form10.Text81.Text = cong(0)
form10.Text66.Text = cong(1)
form10.Text67.Text = cong(2)
form10.Text68.Text = cong(3)
form10.Text69.Text = cong(4)
form10.Text70.Text = cong(5)
form10.Text71.Text = cong(6)
form10.Text72.Text = cong(7)
form10.Text73.Text = cong(8)

```

```
Sub Command1_Click ()
frame3d1.Enabled = True
If OPTION3D1.Value = -1 Then
    Unload FORM15
    FORM15.Show
    FORM15.Text9.Text = "POR DESTINO"
    frame3d1.Enabled = False
    Unload form24
End If
```

```
If OPTION3D2.Value = -1 Then
    Unload FORM15
    FORM15.Show
    FORM15.Text9.Text = "POR RUTAS"
    frame3d1.Enabled = False
    Unload form24
End If
```

End Sub

```
Sub Command2_Click ()
form1.Show
Unload FORM15
Unload form24
```

End Sub

```

Sub Command1_Click ()
Dim db As database
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("carga_procesador")
Dim p, r As Integer
iniciov = form13.MaskedEdit1.Text
finalv = form13.MaskedEdit2.Text
diav = form13.MaskedEdit3.Text
horav = form13.MaskedEdit4.Text

```

```

    naset.AddNew
    naset("hora") = Text2.Text
    naset("pload") = Text3.Text
    naset("offdo") = Text4.Text
    naset("offdi") = Text5.Text
    naset("fichdo") = Text6.Text
    naset("fichdi") = Text7.Text
    naset("inicio") = iniciov
    naset("final") = finalv
    naset("diap") = diav
    naset("horap") = horav
    naset.Update
    MsgBox ("Datos Grabados")
    form29.Show

```

```
End Sub
```

```

Sub Command2_Click ()
form1.Show
form25.Hide
form25.Text2.Text = ""
form25.Text3.Text = ""
form25.Text4.Text = ""
form25.Text5.Text = ""
form25.Text6.Text = ""
form25.Text7.Text = ""

```

```
End Sub
```

```

Sub Form_Load ()
Dim db As database
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("carga_procesador")
iniciov = form13.MaskedEdit1.Text
diapv = form13.MaskedEdit3.Text
label10.Caption = diapv

```

```
End Sub
```

```

Sub Text2_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text2.Text = ""

```

```
End If
```



```
End Sub
```

```
Sub Text2_LostFocus ()
```

```
Dim db As database
```

```
Dim dbnaset As dynaset
```

```
Dim o As Variant
```

```
Set db = OpenDatabase("C:\vb\Pruebas.MDB")
```

```
Set dbnaset = db.CreateDynaset("carga_procesador")
```

```
o = 0
```

```
inicio = form13.MaskedEdit1.Text
```

```
Do While dbnaset.EOF = False
```

```
    If inicio = dbnaset("inicio") Then
```

```
        If Text2.Text = dbnaset("hora") Then
```

```
            o = 1
```

```
        End If
```

```
    End If
```

```
    dbnaset.MoveNext
```

```
Loop
```

```
If o = 1 Then
```

```
    Text2.Text = ""
```

```
    MsgBox ("En esta hora estan ya tomadas las medidas")
```

```
    Text2.SetFocus
```

```
    o = 0
```

```
Else
```

```
    Text3.SetFocus
```

```
End If
```

```
End Sub
```

```
Sub Text3_KeyPress (keyascii As Integer)
```

```
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And  
keyascii <= Asc("Z") Then
```

```
    Beep
```

```
    keyascii = 0
```

```
    Text3.Text = ""
```

```
End If
```

```
End Sub
```

```
Sub Text4_KeyPress (keyascii As Integer)
```

```
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And  
keyascii <= Asc("Z") Then
```

```
    Beep
```

```
    keyascii = 0
```

```
    Text4.Text = ""
```

```
End If
```

```
End Sub
```

```
Sub Text5_KeyPress (keyascii As Integer)
```

```
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And  
keyascii <= Asc("Z") Then
```

```
    Beep
```

```
    keyascii = 0
```

```
    Text5.Text = ""
```

```
End If
```

End Sub

Sub Text6\_KeyPress (keyascii As Integer)

If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And keyascii <= Asc("Z") Then

    Beep

    keyascii = 0

    Text6.Text = ""

End If

End Sub

Sub Text7\_KeyPress (keyascii As Integer)

If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And keyascii <= Asc("Z") Then

    Beep

    keyascii = 0

    Text7.Text = ""

End If

End Sub

---

```
Sub Command1_Click ()  
form26.Hide  
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
form26.Hide
```

```
End Sub
```

```
Sub Command1_Click ()  
FORM1.Show  
Unload form27  
form27.Hide  
End Sub
```

```
Sub Command2_Click ()  
form9.Show  
Unload form27  
form27.Hide  
End Sub
```

```
Sub Command3_Click ()  
form40.Show  
form40.Panel3D1.Visible = True  
form40.Command3.Visible = True  
form40.Command2.Visible = True
```

```
form7.Command3.Visible = True  
form7.Command2.Visible = True  
form27.Hide
```

```
End Sub
```

```

                r = 1
            End If
        End If
        dbnaset.MoveNext
    Loop

    If r = 1 Then
        MsgBox ("En este día estan ya tomadas las medidas")
        MaskedEdit1.SetFocus
    Else
        text5.SetFocus
    End If

End Sub

Sub MaskedEdit1_ValidationError (InvalidText As String, StartPosition As Integer)
MsgBox ("Ingrese P. F. la fecha (Formato Indicado)")
MaskedEdit1.SetFocus
End Sub

Sub Text2_KeyPress (keyascii As Integer)

End Sub

Sub Text2_LostFocus ()

End Sub

Sub Text3_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    text3.Text = ""
End If

End Sub

Sub Text4_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text4.Text = ""
End If

End Sub

Sub Text5_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    text5.Text = ""
End If

End Sub

```

```

                r = 1
            End If
        End If
        dbnaset.MoveNext
    Loop

    If r = 1 Then
        MsgBox ("En este día estan ya tomadas las medidas")
        MaskedEdit1.SetFocus
    Else
        text5.SetFocus
    End If

End Sub

Sub MaskedEdit1_ValidationError (InvalidText As String, StartPosition As Integer)
    MsgBox ("Ingrese P. F. la fecha (Formato Indicado)")
    MaskedEdit1.SetFocus
End Sub

Sub Text2_KeyPress (keyascii As Integer)

End Sub

Sub Text2_LostFocus ()

End Sub

Sub Text3_KeyPress (keyascii As Integer)
    If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
    keyascii <= Asc("Z") Then
        Beep
        keyascii = 0
        text3.Text = ""
    End If
End Sub

Sub Text4_KeyPress (keyascii As Integer)
    If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
    keyascii <= Asc("Z") Then
        Beep
        keyascii = 0
        Text4.Text = ""
    End If
End Sub

Sub Text5_KeyPress (keyascii As Integer)
    If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
    keyascii <= Asc("Z") Then
        Beep
        keyascii = 0
        text5.Text = ""
    End If
End Sub

```

---

```
Sub Text6_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text6.Text = ""
End If

End Sub
```

```
Sub Text7_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text7.Text = ""
End If

End Sub
```

---

```
Sub Command1_Click ()  
form29.Hide  
form25.Text2.Text = ""  
form25.Text3.Text = ""  
form25.Text4.Text = ""  
form25.Text5.Text = ""  
form25.Text6.Text = ""  
form25.Text7.Text = ""
```

```
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
form25.Hide  
form29.Hide  
End Sub
```



---

```
Sub Command1_Click ()  
form41.Show  
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
Unload form38  
Unload form31  
form38.Hide
```

```
End Sub
```

```
Sub Command3_Click ()  
form38.Show  
form31.Hide  
End Sub
```

```
Sub Command1_Click ()  
Unload form28  
form28.Show  
form32.Hide  
Unload form32
```

```
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
form28.Hide  
form32.Hide  
End Sub
```

```
Sub Command1_Click ()
form34.Show
End Sub

Sub Command2_Click ()
Unload form33
form1.Show
form33.Hide
End Sub

Sub Text12_LostFocus ()
text12.Text = UCase$(text12.Text)

Dim db As database
Dim calidad As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set calidad = db.CreateDynaset("calidad_servicio")
Dim p As Variant
p = 1
iniciov = form13.MaskedEdit1.Text
Do While calidad.EOF = False
  If iniciov = calidad("inicio") Then
    If text12.Text = calidad("ruta") Then
      p = 0
    End If
  End If
  calidad.MoveNext
Loop

If p = 0 Then
  text12.Text = ""
  MsgBox ("estos datos ya existen")
  text12.SetFocus
End If

End Sub
```

```

Sub Command1_Click ()
Dim db As database
Dim calidad As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set calidad = db.CreateDynaset("calidad_servicio")
Dim var, var1, var2, var3, p As Variant
p = 1
iniciov = form13.MaskedEdit1.Text
Do While calidad.EOF = False
    If iniciov = calidad("inicio") Then
        If form33.Text12.Text = calidad("ruta") Then
            p = 0
        End If
    End If
    calidad.MoveNext
Loop

If p = 1 Then
    calidad.AddNew
    calidad("ruta") = form33.Text12.Text
    calidad("state") = form33.Text1.Text
    calidad("qta") = form33.Text2.Text
    calidad("overlap") = form33.Text3.Text
    calidad("srs") = form33.Text4.Text
    calidad("regpos") = form33.Text5.Text
    calidad("samples") = form33.Text6.Text
    calidad("fulfil") = form33.Text7.Text
    calidad("qtadate") = form33.Text8.Text
    calidad("qtatime") = Val(form33.Text9.Text)
    calidad("ncdial") = form33.Text10.Text
    calidad("prere") = form33.Text11.Text
    calidad("sidle") = Text1.Text
    calidad("snans") = Text2.Text
    calidad("sans") = Text3.Text
    calidad("bbusy") = Text4.Text
    calidad("unobnr") = Text5.Text

    calidad("icong") = Text6.Text
    calidad("ocong") = Text8.Text
    calidad("tefault") = Text10.Text
    calidad("pdd") = Text7.Text
    calidad("banswer") = Text9.Text
    calidad("ctime") = Text11.Text
    calidad("inicio") = iniciov
    calidad.Update
    MsgBox ("Datos grabados ")
End If

form39.Show

```

End Sub

```
Sub Command2_Click ()  
form1.Show  
Unload form34  
Unload form33  
form34.Hide  
form33.Hide  
End Sub
```

```

Sub Command1_Click ()
Dim db As database
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("Observacion_ trafico")
naset.AddNew
naset("ruta") = form35.Text1.Text
naset("qta") = form35.Text2.Text
naset("obsnr") = form35.Text3.Text
naset("eoscode") = form35.Text4.Text
naset("eosgrp") = form35.Text5.Text
naset("file") = form35.Text6.Text
naset("bnb") = form35.Text7.Text
naset("bnb1") = form35.Text8.Text
naset("anb") = form35.Text9.Text
naset("incdev") = form35.Text10.Text
naset("outdev") = Text1.Text
naset("trd") = Text2.Text
naset("pdd") = Text3.Text
naset("banswer") = Text4.Text
naset.Update
MsgBox ("Datos Grabados")
form1.Show
Unload form35
Unload form36
End Sub

Sub Command2_Click ()
form1.Show
Unload form35
Unload form36
End Sub

Sub Text1_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text1.Text = ""
End If
End Sub

Sub Text3_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text3.Text = ""
End If

End Sub

Sub Text4_KeyPress (keyascii As Integer)
If keyascii >= Asc("a") And keyascii <= Asc("z") Or keyascii = 255 Or keyascii >= Asc("A") And
keyascii <= Asc("Z") Then
    Beep
    keyascii = 0
    Text4.Text = ""
End If

End Sub

```

```
Sub Command1_Click ()

form1.Show
Unload form37
Unload form9
form37.Hide

End Sub

Sub Command2_Click ()
form27.Show
Unload form37
form37.Hide
End Sub

Sub Command3_Click ()
'construye la grafica para obtener el trafico saliente en hora pico
Dim db As database
Dim dnaset As dynaset
Dim v As Variant
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("trafico_saliente")
iniciov = form13.MaskedEdit1.Text
Dim var As Variant
Do While dnaset.EOF = False
    If iniciov = dnaset("inicio") Then
        var = dnaset("horap")
        form42.Graph2.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
        form42.Graph2.GraphData = dnaset("traff")
    End If
    dnaset.MoveNext
Loop

form42.Show
form37.Hide
End Sub

Sub Form_Load ()
combo1.Text = ""
combo1.AddItem "CAYO"
combo1.AddItem "STDO"
combo1.AddItem "SGQI"
combo1.AddItem "CAYI"

End Sub
```

---

```
Sub Command1_Click ()  
form41.Show  
End Sub
```

```
Sub Command2_Click ()  
form1.Show  
Unload form38  
Unload form31  
form38.Hide
```

```
End Sub
```

```
Sub Command3_Click ()  
form31.Show  
form38.Hide
```

```
End Sub
```



---

```
Sub Command1_Click()  
form1.Visible = True  
form4.Visible = False  
End Sub
```

```
Sub Command1_Click ()
ESTE PROCEDIMIENTO OBTIENE LA GRAFICA DE CALIDAD DE SERVICIO DE ACUERDO A
LOS
PARAMETROS QUE TENGAN VALORES EN ESE MOMENTO, POR LO MENOS DOS
Dim db As database
Dim dnaset As dynaset
Dim var As Variant
Dim v As Variant
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("calidad_servicio")
iniciov = form13.MaskedEdit1.Text
Dim ban As Integer
ban = 9
Do While dnaset.EOF = False

    If iniciov = dnaset("inicio") Then
        If text2.Text = dnaset("ruta") Then

            If dnaset("ncdial") = 0 Then
                ban = ban - 1
            End If

            If dnaset("prere") = 0 Then
                ban = ban - 1
            End If

            If dnaset("snars") = 0 Then
                ban = ban - 1
            End If

            If dnaset("sans") = 0 Then
                ban = ban - 1
            End If

            If dnaset("bbusy") = 0 Then
                ban = ban - 1
            End If

            If dnaset("unobor") = 0 Then
                ban = ban - 1
            End If

            If dnaset("icong") = 0 Then
                ban = ban - 1
            End If

            If dnaset("ocong") = 0 Then
                ban = ban - 1
            End If

            If dnaset("tefault") = 0 Then
                ban = ban - 1
            End If

form7.Graph1.NumPoints = ban
```

```
If dnaset("ncdial") = 0 And dnaset("prere") <> 0 And dnaset("snans") <> 0 And dnaset("sans") <> 0 And dnaset("bbusy") <> 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And dnaset("tefault") <> 0 Then
```

```
form7.Graph1.GraphData = dnaset("prere")
form7.Graph1.GraphData = dnaset("snans")
form7.Graph1.GraphData = dnaset("sans")
form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Repone"
form7.Graph1.LegendText = "B No contesta "
form7.Graph1.LegendText = "B Contesta"
form7.Graph1.LegendText = "B Ocupado"
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Congestión Externa"
form7.Graph1.LegendText = "Falla Técnica"
```

```
End If
```

```
If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") <> 0 And dnaset("sans") <> 0 And dnaset("bbusy") <> 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And dnaset("tefault") <> 0 Then
```

```
form7.Graph1.GraphData = dnaset("snans")
form7.Graph1.GraphData = dnaset("sans")
form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B No contesta "
form7.Graph1.LegendText = "B Contesta"
form7.Graph1.LegendText = "B Ocupado"
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Congestión Externa"
form7.Graph1.LegendText = "Falla Técnica"
```

```
End If
```

```
If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") <> 0 And dnaset("bbusy") <> 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And dnaset("tefault") <> 0 Then
```

```
form7.Graph1.GraphData = dnaset("sans")
form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Contesta"
form7.Graph1.LegendText = "B Ocupado"
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Congestión Externa"
```

```

    form7.Graph1.LegendText = "Falla Técnica"
  End If

```

```

  If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0
  And dnaset("bbusy") <> 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <>
  0 And dnaset("tefault") <> 0 Then

```

```

    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("unobnr")
    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "B Ocupado"
    form7.Graph1.LegendText = "B Inobtenible"
    form7.Graph1.LegendText = "Congestión Interna"
    form7.Graph1.LegendText = "Congestión Externa"
    form7.Graph1.LegendText = "Falla Técnica"

```

```

  End If

```

```

  If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
  dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
  dnaset("tefault") <> 0 Then

```

```

    form7.Graph1.GraphData = dnaset("unobnr")
    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "B Inobtenible"
    form7.Graph1.LegendText = "Congestión Interna"
    form7.Graph1.LegendText = "Congestión Externa"
    form7.Graph1.LegendText = "Falla Técnica"

```

```

  End If

```

```

  If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
  dnaset("bbusy") = 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
  dnaset("tefault") <> 0 Then

```

```

    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "Congestión Interna"
    form7.Graph1.LegendText = "Congestión Externa"
    form7.Graph1.LegendText = "Falla Técnica"

```

```

  End If

```

'comienza los datos en orden

```

  If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
  dnaset("bbusy") = 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
  dnaset("tefault") <> 0 Then

```

```

    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "Congestión Externa"
    form7.Graph1.LegendText = "Falla Técnica"

```

```

  End If

```

```

  If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
  dnaset("bbusy") = 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
  dnaset("tefault") <> 0 Then

```

```

form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "Congestión interna"
form7.Graph1.LegendText = "Falla Técnica"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.LegendText = "Congestión interna"
form7.Graph1.LegendText = "Congestión Externa"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
dnaset("tefault") <> 0 Then
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "Congestión interna"
form7.Graph1.LegendText = "Congestión Externa"
form7.Graph1.LegendText = "Falla Técnica"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Externa"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") <> 0 Then
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Externa"
form7.Graph1.LegendText = "Falla Técnica"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
dnaset("tefault") = 0 Then
form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"

End If

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
dnaset("tefault") <> 0 Then
form7.Graph1.GraphData = dnaset("unobnr")

```

```

form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Falla Técnica"

```

End If

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then

```

```

form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Congestión Externa"

```

End If

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") = 0 And dnaset("unobnr") <> 0 And dnaset("icong") <> 0 And dnaset("ocong") <> 0 And
dnaset("tefault") <> 0 Then

```

```

form7.Graph1.GraphData = dnaset("unobnr")
form7.Graph1.GraphData = dnaset("icong")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Inobtenible"
form7.Graph1.LegendText = "Congestión Interna"
form7.Graph1.LegendText = "Congestión Externa"
form7.Graph1.LegendText = "Falla Técnica"

```

End If

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") = 0 And
dnaset("tefault") <> 0 Then

```

```

form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("tefault")
form7.Graph1.LegendText = "B Ocupados"
form7.Graph1.LegendText = "Falla Técnica"

```

End If

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then

```

```

form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.LegendText = "B Ocupados"
form7.Graph1.LegendText = "Congestión Externa"

```

End If

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then

```

```

form7.Graph1.GraphData = dnaset("bbusy")
form7.Graph1.GraphData = dnaset("ocong")
form7.Graph1.LegendText = "B Ocupados"

```

```

    form7.Graph1.LegendText = "Congestión Externa"
End If

```

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") = 0 Then
    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.LegendText = "B Ocupados"
    form7.Graph1.LegendText = "Congestión Externa"
End If

```

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") = 0 And dnaset("ocong") <> 0 And
dnaset("tefault") <> 0 Then
    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("ocong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "B Ocupados"
    form7.Graph1.LegendText = "Congestión Externa"
    form7.Graph1.LegendText = "Falla Técnica"
End If

```

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
dnaset("tefault") = 0 Then
    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.LegendText = "B Ocupados"
    form7.Graph1.LegendText = "Congestión Interna"
End If

```

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
dnaset("tefault") = 0 Then
    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.LegendText = "B Ocupados"
    form7.Graph1.LegendText = "Congestión Interna"
End If

```

```

If dnaset("ncdial") = 0 And dnaset("prere") = 0 And dnaset("snans") = 0 And dnaset("sans") = 0 And
dnaset("bbusy") <> 0 And dnaset("unobnr") = 0 And dnaset("icong") <> 0 And dnaset("ocong") = 0 And
dnaset("tefault") <> 0 Then
    form7.Graph1.GraphData = dnaset("bbusy")
    form7.Graph1.GraphData = dnaset("icong")
    form7.Graph1.GraphData = dnaset("tefault")
    form7.Graph1.LegendText = "B Ocupados"
    form7.Graph1.LegendText = "Congestión Interna"
    form7.Graph1.LegendText = "Falla Técnica"
End If

```

```

Sub Command1_Click ()
Dim db As database
Dim naset As dynaset
Dim saliente As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set naset = db.CreateDynaset("carga_procesador_dia")
Set saliente = db.CreateDynaset("trafico_Saliente")
iniciov = form13.MaskedEdit1.Text
Dim i As Integer
i = 0
Dim var As Variant
Dim fin As Variant
ReDim dia(31) As Variant
ReDim plod(31) As Variant
ReDim llam(31) As Variant
ReDim traf(31) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        fin = naset("final")
        dia(i) = Val(Mid$(naset("dia"), 1, 2))
        plod(i) = naset("pload")
        llam(i) = naset("flchdi")
        i = i + 1
    End If
    naset.MoveNext
Loop
i = 0
Do While saliente.EOF = False
    If iniciov = saliente("inicio") Then
        traf(i) = saliente("traff")
        i = i + 1
    End If
    saliente.MoveNext
Loop
i = 0
Set naset = db.CreateDynaset("carga_procesador_dia")
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If 1 = Val(Mid$(naset("dia"), 1, 2)) Then
            naset.Edit
            naset("pload") = form31.Text18.Text
            naset("flchdi") = form31.Text39.Text
            naset.Update
        End If

        If 2 = Val(Mid$(naset("dia"), 1, 2)) Then
            naset.Edit
            naset("pload") = form31.Text19.Text
            naset("flchdi") = form31.Text40.Text
            naset.Update
        End If

        If 3 = Val(Mid$(naset("dia"), 1, 2)) Then
            naset.Edit
            naset("pload") = form31.Text20.Text
            naset("flchdi") = form31.Text41.Text
        End If
    End If
    naset.MoveNext
Loop

```



```
naset.Update
End If
If 4 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text21.Text
    naset("fchdi") = form31.Text42.Text
    naset.Update
End If
If 5 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text22.Text
    naset("fchdi") = form31.Text43.Text
    naset.Update
End If
If 6 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text23.Text
    naset("fchdi") = form31.Text44.Text
    naset.Update
End If
If 7 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text24.Text
    naset("fchdi") = form31.Text45.Text
    naset.Update
End If
If 8 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text25.Text
    naset("fchdi") = form31.Text46.Text
    naset.Update
End If
If 9 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text26.Text
    naset("fchdi") = form31.Text47.Text
    naset.Update
End If
If 10 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text27.Text
    naset("fchdi") = form31.Text48.Text
    naset.Update
End If
If 11 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text28.Text
    naset("fchdi") = form31.Text49.Text
    naset.Update
End If
If 12 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text29.Text
    naset("fchdi") = form31.Text50.Text
    naset.Update
End If
If 13 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text30.Text
    naset("fchdi") = form31.Text51.Text
```

```
naset.Update
End If
If 14 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form31.Text31.Text
    naset("fchdi") = form31.Text52.Text
    naset.Update
End If
If 15 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text18.Text
    naset("fchdi") = form38.Text52.Text
    naset.Update
End If
If 16 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text19.Text
    naset("fchdi") = form38.Text53.Text
    naset.Update
End If
If 17 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text20.Text
    naset("fchdi") = form38.Text54.Text
    naset.Update
End If
If 18 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text21.Text
    naset("fchdi") = form38.Text55.Text
    naset.Update
End If
If 19 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text22.Text
    naset("fchdi") = form38.Text56.Text
    naset.Update
End If
If 20 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text23.Text
    naset("fchdi") = form38.Text57.Text
    naset.Update
End If
If 21 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text24.Text
    naset("fchdi") = form38.Text58.Text
    naset.Update
End If
If 22 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text25.Text
    naset("fchdi") = form38.Text59.Text
    naset.Update
End If

If 23 = Val(Mid$(naset("dia"), 1, 2)) Then
    naset.Edit
    naset("pload") = form38.Text26.Text
```

```

        nset("fchdi") = form38.Text60.Text
    nset.Update
End If
If 24 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text27.Text
    nset("fchdi") = form38.Text61.Text
    nset.Update
End If
If 25 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text28.Text
    nset("fchdi") = form38.Text62.Text
    nset.Update
End If
If 26 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text29.Text
    nset("fchdi") = form38.Text63.Text
    nset.Update
End If
If 27 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text30.Text
    nset("fchdi") = form38.Text64.Text
    nset.Update
End If
If 28 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text31.Text
    nset("fchdi") = form38.Text65.Text
    nset.Update
End If
If 29 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text32.Text
    nset("fchdi") = form38.Text66.Text
    nset.Update
End If
If 30 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text33.Text
    nset("fchdi") = form38.Text67.Text
    nset.Update
End If
If 31 = Val(Mid$(nset("dia"), 1, 2)) Then
    nset.Edit
    nset("pload") = form38.Text34.Text
    nset("fchdi") = form38.Text68.Text
    nset.Update
End If

End If
nset.MoveNext

```

Loop

Set saliente = db.CreateDynaset("trafico\_Saliente")

```
Do While saliente.EOF = False
  If iniciov = saliente("inicio") Then
    If 1 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text56.Text
      saliente.Update
    End If
    If 2 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text53.Text
      saliente.Update
    End If
    If 3 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text54.Text
      saliente.Update
    End If
    If 4 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text55.Text
      saliente.Update
    End If
    If 5 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text32.Text
      saliente.Update
    End If
    If 6 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text33.Text
      saliente.Update
    End If
    If 7 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text34.Text
      saliente.Update
    End If
    If 8 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text35.Text
      saliente.Update
    End If
    If 9 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text36.Text
      saliente.Update
    End If
    If 10 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text37.Text
      saliente.Update
    End If
    If 11 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text38.Text
      saliente.Update
    End If
    If 12 = Val(Mid$(saliente("dia"), 1, 2)) Then
      saliente.Edit
      saliente("traff") = form31.Text1.Text
    End If
  End If
Loop
```

```
        saliente.Update
    End If
    If 13 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form31.Text2.Text
        saliente.Update
    End If
    If 14 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form31.Text3.Text
        saliente.Update
    End If
    If 15 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text35.Text
        saliente.Update
    End If
    If 16 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text36.Text
        saliente.Update
    End If
    If 17 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text37.Text
        saliente.Update
    End If
    If 18 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text38.Text
        saliente.Update
    End If
    If 19 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text39.Text
        saliente.Update
    End If
    If 20 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text40.Text
        saliente.Update
    End If
    If 21 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text41.Text
        saliente.Update
    End If
    If 22 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text42.Text
        saliente.Update
    End If
    If 23 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text43.Text
        saliente.Update
    End If
    If 24 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text44.Text
```

```
        saliente.Update
    End If
    If 25 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text45.Text
        saliente.Update
    End If
    If 26 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text46.Text
        saliente.Update
    End If
    If 27 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text47.Text
        saliente.Update
    End If
    If 28 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text48.Text
        saliente.Update
    End If
    If 29 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text49.Text
        saliente.Update
    End If
    If 30 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text50.Text
        saliente.Update
    End If
    If 31 = Val(Mid$(saliente("dia"), 1, 2)) Then
        saliente.Edit
        saliente("traff") = form38.Text51.Text
        saliente.Update
    End If

    End If
    saliente.MoveNext
Loop

MsgBox ("Datos Grabados....")

form1.Show
form31.Hide
End Sub

Sub Command2_Click ()
form41.Hide
End Sub
```

```
Sub Command1_Click ()  
form1.Show  
Unload form42  
Unload form9
```

```
form42.Hide  
End Sub
```

```
Sub Command2_Click ()  
form7.Show  
Unload form42  
form42.Hide  
End Sub
```

```
Sub Command1_Click ()
Unload FORM43
Unload FORM23
Unload FORM10
FORM1.Show
End Sub

Sub Command3_Click ()
Dim db As database
Dim dat As dynaset
Dim sal, ert As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

Unload FORM43
Unload FORM10

form44.Text5.Text = 27
form44.Text6.Text = 28
form44.Text7.Text = 29
form44.Text8.Text = 30
form44.Text9.Text = 31
form44.Text10.Text = 32
form44.Text18.Text = 32

form44.Text79.Text = VEC(26)
form44.Text80.Text = VEC(27)
form44.Text81.Text = VEC(28)
form44.Text1.Text = VEC(29)
form44.Text2.Text = VEC(30)
form44.Text3.Text = VEC(31)

form44.Text12.Text = abr(26)
form44.Text13.Text = abr(27)
form44.Text14.Text = abr(28)
form44.Text15.Text = abr(29)
form44.Text16.Text = abr(30)
form44.Text17.Text = abr(31)
```



```
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Set naset = db.CreateDynaset("Sis_Perdida")
Do While naset.EOF = False
```

```
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            var(i) = naset("ndv")
            i = i + 1
```

```
        End If
    End If
    naset.MoveNext
Loop
```

```
Dim sum As Variant
sum = 0
i = 0
For i = 0 To 35
```

```
    sum = sum + var(i)
```

```
Next i
```

```
Set naset = db.CreateDynaset("sis_perdida")
i = 0
ReDim bloq(35) As Variant
Do While naset.EOF = False
```

```
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            bloq(i) = naset("anblo")
            i = i + 1
```

```
        End If
    End If
    naset.MoveNext
```

```
Loop
```

```
Dim sum1 As Variant
sum1 = 0
i = 0
For i = 0 To 35
```

```
    sum1 = sum1 + bloq(i)
```

```
Next i
```

```
Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
```

```

If iniciov = naset("inicio") Then

    If abr(i) + "O" = naset("ruta") Then
        saliente(i) = naset("traff")
        i = i + 1
    End If

    If abr(j) + "I" = naset("ruta") Then
        entrante(j) = naset("traff")
        j = j + 1
    End If
End If
naset.MoveNext
Loop

Dim sum2, sum3 As Variant
sum2 = 0
sum3 = 0
i = 0
For i = 0 To 35

    sum2 = sum2 + entrante(i)
    sum3 = sum3 + saliente(i)

Next i

Set naset = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            cong(i) = naset("ccong")
            i = i + 1
        End If
    End If
    naset.MoveNext

Loop
Dim c As Variant

c = 0
i = 0
For i = 0 To 35

    c = c + cong(i)

Next i

' halla la suma de traficos entrantes y salientes
ReDim total(35) As Variant
Dim tot As Variant
i = 0
For i = 0 To 35

```

```
total(i) = saliente(i) + entrante(i)
tot = tot + total(i)
Next i
```

```
form44.Text54.Text = total(26)
form44.Text55.Text = total(27)
form44.Text56.Text = total(28)
form44.Text57.Text = total(29)
form44.Text58.Text = total(30)
form44.Text59.Text = total(31)
form44.Text60.Text = tot
```

```
form44.Text19.Text = var(26)
form44.Text20.Text = var(27)
form44.Text21.Text = var(28)
form44.Text22.Text = var(29)
form44.Text23.Text = var(30)
form44.Text24.Text = var(31)
form44.Text25.Text = sum
```

```
form44.Text72.Text = cong(26)
form44.Text73.Text = cong(27)
form44.Text74.Text = cong(28)
form44.Text75.Text = cong(29)
form44.Text76.Text = cong(30)
form44.Text77.Text = cong(31)
form44.Text78.Text = c
```

```
form44.Text47.Text = saliente(26)
form44.Text48.Text = saliente(27)
form44.Text49.Text = saliente(28)
form44.Text50.Text = saliente(29)
form44.Text51.Text = saliente(30)
form44.Text52.Text = saliente(31)
form44.Text53.Text = sum3
```

```
form44.Text33.Text = bloq(26)
form44.Text34.Text = bloq(27)
form44.Text35.Text = bloq(28)
form44.Text36.Text = bloq(29)
form44.Text37.Text = bloq(30)
form44.Text38.Text = bloq(31)
form44.Text39.Text = sum1
'para mostrar datos de traff
```

```
form44.Text40.Text = entrante(26)
form44.Text41.Text = entrante(27)
form44.Text42.Text = entrante(28)
```

```

form44.Text43.Text = entrante(29)
form44.Text44.Text = entrante(30)
form44.Text45.Text = entrante(31)
form44.Text46.Text = sum2
para mostrar ccong

```

```

Unload FORM43
form44.Show

```

```
End Sub
```

```

Sub Command4_Click ()
Unload FORM43
Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
inicio = form13.MaskedEdit1.Text
Unload FORM10
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

ReDim ocupent(35) As Variant

ReDim ocupsal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If inicio = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            ocupsal(i) = naset("mhtime")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            ocupent(j) = naset("mhtime")
            j = j + 1
        End If
    End If

```

```

    End If
    naset.MoveNext
Loop

Set naset = db.CreateDynaset("Sis_Perdida")
ReDim intent(35) As Variant

ReDim intsal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            intsal(i) = naset("nbids")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            intent(j) = naset("nbids")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop

```

```

Set naset = db.CreateDynaset("Sis_Perdida")
ReDim conent(35) As Variant

ReDim consal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            consal(i) = naset("nbansw")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            conent(j) = naset("nbansw")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop

```

```

ReDim pcent(35) As Double
ReDim pcsal(35) As Variant
Dim a, b As Variant
Dim c, d, j1 As Variant

For i = 0 To 31

    a = conent(i) * 100
    b = consal(i) * 100

```

```
If intent(i) <> 0 Then
  c = a / intent(i)
  d = b / intsal(i)
Else
  c = 0
  d = 0
End If
pcent(i) = c
pcsal(i) = d
Next i
```

```
form45.Text151.Text = pcent(9)
form45.Text133.Text = pcent(10)
form45.Text135.Text = pcent(11)
form45.Text137.Text = pcent(12)
form45.Text110.Text = pcent(13)
form45.Text111.Text = pcent(14)
form45.Text113.Text = pcent(15)
form45.Text115.Text = pcent(16)
form45.Text117.Text = pcent(17)
form45.Text119.Text = pcent(18)
form45.Text121.Text = pcent(19)
form45.Text123.Text = pcent(20)
form45.Text125.Text = pcent(21)
form45.Text127.Text = pcent(22)
form45.Text129.Text = pcent(23)
form45.Text99.Text = pcent(24)
form45.Text100.Text = pcent(25)
```

```
form45.Text153.Text = pcsal(9)
form45.Text147.Text = pcsal(10)
form45.Text148.Text = pcsal(11)
form45.Text149.Text = pcsal(12)
form45.Text150.Text = pcsal(13)
form45.Text112.Text = pcsal(14)
form45.Text114.Text = pcsal(15)
form45.Text116.Text = pcsal(16)
form45.Text118.Text = pcsal(17)
```

```
form45.Text120.Text = pcsal(18)
form45.Text122.Text = pcsal(19)
form45.Text124.Text = pcsal(20)
form45.Text126.Text = pcsal(21)
form45.Text128.Text = pcsal(22)
form45.Text130.Text = pcsal(23)
form45.Text131.Text = pcsal(24)
form45.Text132.Text = pcsal(25)
```

form45.Text47.Text = ocupent(9)  
form45.Text48.Text = ocupent(10)  
form45.Text49.Text = ocupent(11)  
form45.Text50.Text = ocupent(12)  
form45.Text51.Text = ocupent(13)  
form45.Text52.Text = ocupent(14)  
form45.Text53.Text = ocupent(15)  
form45.Text54.Text = ocupent(16)  
form45.Text55.Text = ocupent(17)

form45.Text56.Text = ocupent(18)  
form45.Text57.Text = ocupent(19)  
form45.Text58.Text = ocupent(20)  
form45.Text59.Text = ocupent(21)  
form45.Text60.Text = ocupent(22)  
form45.Text61.Text = ocupent(23)  
form45.Text62.Text = ocupent(24)  
form45.Text63.Text = ocupent(25)

form45.Text30.Text = ocupsal(9)  
form45.Text31.Text = ocupsal(10)  
form45.Text32.Text = ocupsal(11)  
form45.Text33.Text = ocupsal(12)  
form45.Text34.Text = ocupsal(13)  
form45.Text35.Text = ocupsal(14)  
form45.Text36.Text = ocupsal(15)  
form45.Text37.Text = ocupsal(16)  
form45.Text38.Text = ocupsal(17)

form45.Text39.Text = ocupsal(18)  
form45.Text40.Text = ocupsal(19)  
form45.Text41.Text = ocupsal(20)  
form45.Text42.Text = ocupsal(21)  
form45.Text43.Text = ocupsal(22)  
form45.Text44.Text = ocupsal(23)  
form45.Text45.Text = ocupsal(24)  
form45.Text46.Text = ocupsal(25)

form45.Text13.Text = intent(9)  
form45.Text14.Text = intent(10)  
form45.Text15.Text = intent(11)  
form45.Text16.Text = intent(12)  
form45.Text17.Text = intent(13)  
form45.Text18.Text = intent(14)  
form45.Text19.Text = intent(15)  
form45.Text20.Text = intent(16)  
form45.Text21.Text = intent(17)

form45.Text22.Text = intent(18)  
form45.Text23.Text = intent(19)  
form45.Text24.Text = intent(20)  
form45.Text25.Text = intent(21)  
form45.Text26.Text = intent(22)  
form45.Text27.Text = intent(23)  
form45.Text28.Text = intent(24)

form45.Text29.Text = intent(25)

form45.Text143.Text = intsal(9)  
form45.Text144.Text = intsal(10)  
form45.Text145.Text = intsal(11)  
form45.Text146.Text = intsal(12)  
form45.Text86.Text = intsal(13)  
form45.Text1.Text = intsal(14)  
form45.Text2.Text = intsal(15)  
form45.Text3.Text = intsal(16)  
form45.Text4.Text = intsal(17)

form45.Text5.Text = intsal(18)  
form45.Text6.Text = intsal(19)  
form45.Text7.Text = intsal(20)  
form45.Text8.Text = intsal(21)  
form45.Text9.Text = intsal(22)  
form45.Text10.Text = intsal(23)  
form45.Text11.Text = intsal(24)  
form45.Text12.Text = intsal(25)

form45.Text139.Text = conent(9)  
form45.Text140.Text = conent(10)  
form45.Text141.Text = conent(11)  
form45.Text142.Text = conent(12)  
form45.Text101.Text = conent(13)  
form45.Text87.Text = conent(14)  
form45.Text88.Text = conent(15)  
form45.Text89.Text = conent(16)  
form45.Text90.Text = conent(17)

form45.Text91.Text = conent(18)  
form45.Text92.Text = conent(19)  
form45.Text93.Text = conent(20)  
form45.Text94.Text = conent(21)  
form45.Text95.Text = conent(22)  
form45.Text96.Text = conent(23)  
form45.Text97.Text = conent(24)  
form45.Text98.Text = conent(25)

form45.Text152.Text = consal(9)  
form45.Text134.Text = consal(10)  
form45.Text136.Text = consal(11)  
form45.Text138.Text = consal(12)  
form45.Text109.Text = consal(13)  
form45.Text102.Text = consal(14)  
form45.Text103.Text = consal(15)  
form45.Text104.Text = consal(16)  
form45.Text105.Text = consal(17)

form45.Text106.Text = consal(18)  
form45.Text107.Text = consal(19)  
form45.Text108.Text = consal(20)



```
form45.Text81.Text = consal(21)
form45.Text82.Text = consal(22)
form45.Text83.Text = consal(23)
form45.Text84.Text = consal(24)
form45.Text85.Text = consal(25)
```

```
form45.Text64.Text = 10
form45.Text65.Text = 11
form45.Text66.Text = 12
form45.Text67.Text = 13
form45.Text68.Text = 14
form45.Text69.Text = 15
form45.Text70.Text = 16
form45.Text71.Text = 17
form45.Text72.Text = 18
form45.Text73.Text = 19
form45.Text74.Text = 20
form45.Text75.Text = 21
form45.Text76.Text = 22
form45.Text77.Text = 23
form45.Text78.Text = 24
form45.Text79.Text = 25
form45.Text80.Text = 26
```

```
form45.Show
```

```
End Sub
```

```
Sub Command5_Click ()
Unload form31
Unload form38
```

```
Unload form1
Unload form2
Unload form3
Unload form4
Unload form5
Unload form6
Unload form7
Unload form8
Unload form9
Unload form11
Unload form12
Unload form13
Unload form14
Unload form15
Unload form16
Unload form17
Unload form18
```

INSTRUCCIONES NECESARIAS PARA LA ELABORACION DE LOS  
'CUADROS ESTADISTICOS, EN EL QUE SE DETALLARAN, DATOS COMO:  
'NOMBRE DE LA CENTRAL, CIRCUITOS BIDIREC. BLOQUEOS EN HP,  
'CARGA DE PROCESADOR EN HORA PICO ETC.

```
Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
```

```
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop
Unload FORM43
FORM10.Text90.Text = 1
FORM10.Text82.Text = 2
FORM10.Text83.Text = 3
FORM10.Text84.Text = 4
FORM10.Text85.Text = 5
FORM10.Text86.Text = 6
FORM10.Text87.Text = 7
FORM10.Text88.Text = 8
FORM10.Text89.Text = 9

FORM10.Text1.Text = VEC(0)
FORM10.Text2.Text = VEC(1)
FORM10.Text3.Text = VEC(2)
FORM10.Text4.Text = VEC(3)
FORM10.Text5.Text = VEC(4)
FORM10.Text6.Text = VEC(5)
FORM10.Text7.Text = VEC(6)
FORM10.Text8.Text = VEC(7)
FORM10.Text11.Text = VEC(8)

FORM10.Text12.Text = abr(0)
FORM10.Text13.Text = abr(1)
FORM10.Text14.Text = abr(2)
FORM10.Text15.Text = abr(3)
FORM10.Text16.Text = abr(4)
FORM10.Text17.Text = abr(5)
FORM10.Text18.Text = abr(6)
FORM10.Text10.Text = abr(7)
FORM10.Text9.Text = abr(8)
DATOS PARA LA FORMA 43

i = 0
```

BLOQUE PARA HALLAR EL VALOR DE CIRCUITOS BIDIRECCIONALES POR RUTA  
LOS QUE ESTAN EN LA TABLA SIS\_PERDIDA"

Set dat = db.CreateDynaset("sis\_perdida")

ReDim var(35) As Variant

ReDim var1(35) As Variant

i = 0

Do While dat.EOF = False

    If iniciov = dat("inicio") Then

        If abr(i) + "O" = dat("ruta") Then

            var(i) = dat("ndv")

            i = i + 1

        End If

    End If

    dat.MoveNext

Loop

ReDim bloq(35) As Variant

Set dat = db.CreateDynaset("sis\_perdida")

i = 0

Do While dat.EOF = False

    If iniciov = dat("inicio") Then

        If abr(i) + "O" = dat("ruta") Then

            bloq(i) = dat("anblo")

            i = i + 1

        End If

    End If

    dat.MoveNext

Loop

FORM10.Text27.Text = var(0)

FORM10.Text19.Text = var(1)

FORM10.Text20.Text = var(2)

FORM10.Text21.Text = var(3)

FORM10.Text22.Text = var(4)

FORM10.Text23.Text = var(5)

FORM10.Text24.Text = var(6)

FORM10.Text25.Text = var(7)

FORM10.Text26.Text = var(8)

'FINAL DEL PRIMER BLOQUE

'SEGUNDO BLOQUE

FORM10.Text37.Text = bloq(0)

FORM10.Text38.Text = bloq(1)

FORM10.Text39.Text = bloq(2)

FORM10.Text44.Text = bloq(3)

FORM10.Text45.Text = bloq(4)

FORM10.Text40.Text = bloq(5)

FORM10.Text46.Text = bloq(6)

FORM10.Text47.Text = bloq(7)

FORM10.Text48.Text = bloq(8)

'instrucciones para hallar el trafico los valores del trafico saliente  
'y entrante y determinar la suma

Set dat = db.CreateDynaset("sis\_perdida")

i = 0

j = 0

ReDim entrante(35) As Variant

ReDim saliente(35) As Variant

```

ReDim total(35) As Variant
Do While dat.EOF = False
  If iniciov = dat("inicio") Then

    If abr(i) + "O" = dat("ruta") Then
      saliente(i) = dat("traff")
      i = i + 1
    End If

    If abr(j) + "I" = dat("ruta") Then
      entrante(j) = dat("traff")
      j = j + 1
    End If
  End If
  dat.MoveNext
Loop

```

```

'instrucciones para hallar el total
For i = 0 To 35
  total(i) = saliente(i) + entrante(i)
Next i

```

bloque para visualizar los valores de trafico entrante y saliente

```

FORM10.Text42.Text = entrante(0)
FORM10.Text43.Text = entrante(1)
FORM10.Text49.Text = entrante(2)
FORM10.Text50.Text = entrante(3)
FORM10.Text51.Text = entrante(4)
FORM10.Text52.Text = entrante(5)
FORM10.Text53.Text = entrante(6)
FORM10.Text54.Text = entrante(7)
FORM10.Text55.Text = entrante(8)

```

```

FORM10.Text63.Text = saliente(0)
FORM10.Text62.Text = saliente(1)
FORM10.Text61.Text = saliente(2)
FORM10.Text60.Text = saliente(3)
FORM10.Text59.Text = saliente(4)
FORM10.Text58.Text = saliente(5)
FORM10.Text57.Text = saliente(6)
FORM10.Text56.Text = saliente(7)
FORM10.Text41.Text = saliente(8)

```

```

FORM10.Text65.Text = total(0)
FORM10.Text64.Text = total(1)
FORM10.Text74.Text = total(2)
FORM10.Text75.Text = total(3)
FORM10.Text76.Text = total(4)
FORM10.Text77.Text = total(5)
FORM10.Text78.Text = total(6)
FORM10.Text79.Text = total(7)
FORM10.Text80.Text = total(8)

```

halla el valor de porcentaje de congestión de tráfico saliente

```
Set dat = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While dat.EOF = False
    If iniciov = dat("inicio") Then
        If abr(i) + "O" = dat("ruta") Then
            cong(i) = dat("ccong")
            i = i + 1
        End If
    End If
    dat.MoveNext
```

Loop

```
FORM10.Text81.Text = cong(0)
FORM10.Text66.Text = cong(1)
FORM10.Text67.Text = cong(2)
FORM10.Text68.Text = cong(3)
FORM10.Text69.Text = cong(4)
FORM10.Text70.Text = cong(5)
FORM10.Text71.Text = cong(6)
FORM10.Text72.Text = cong(7)
FORM10.Text73.Text = cong(8)
```

```
FORM10.Show
Unload FORM43
End Sub
```

```

Sub Command2_Click ()
form1.Show
form44.Hide
Unload form44
Unload form10
Unload form43

End Sub

Sub Command3_Click ()
Unload form44
Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
inicio = form13.MaskedEdit1.Text
Unload form10
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

ReDim ocupent(35) As Variant

ReDim ocupsal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If inicio = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            ocupsal(i) = naset("mhtime")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            ocupent(j) = naset("mhtime")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop
Dim oc As Variant
Dim oc1 As Variant
oc = 0
oc1 = 0

```

```

For i = 0 To 35
    oc = oc + ocupsal(i)
    oc1 = oc1 + ocupent(i)
Next i

```

```

Set naset = db.CreateDynaset("Sis_Perdida")
ReDim intent(35) As Variant

```

```

ReDim intsal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            intsal(i) = naset("nbids")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            intent(j) = naset("nbids")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop

```

```

Dim in As Variant
Dim in1 As Variant

```

```

in = 0
in1 = 0
For i = 0 To 35
    in = in + intsal(i)
    in1 = in1 + intent(i)
Next i

```

```

Set naset = db.CreateDynaset("Sis_Perdida")
ReDim conent(35) As Variant

```

```

ReDim consal(35) As Variant
i = 0
j = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            consal(i) = naset("nbansw")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            conent(j) = naset("nbansw")
            j = j + 1
        End If
    End If
End If

```

```
naset.MoveNext  
Loop
```

```
Dim cn As Variant  
Dim cn1 As Variant
```

```
cn = 0  
cn1 = 0  
For i = 0 To 35  
    cn = cn + consal(i)  
    cn1 = cn1 + conent(i)  
Next i
```

```
ReDim pcent(35) As Double  
ReDim pcsal(35) As Variant  
Dim a, b As Variant  
Dim c, d As Variant  
Dim x, y As Variant  
x = 0  
y = 0  
For i = 0 To 31
```

```
    a = conent(i) * 100  
    b = consal(i) * 100  
    If intent(i) <> 0 Then  
        c = a / intent(i)  
        d = b / intsal(i)  
    Else  
        c = 0  
        d = 0  
    End If  
    pcent(i) = c  
    pcsal(i) = d  
    x = x + pcent(i)  
    y = y + pcsal(i)
```

```
Next i
```

```
form46.Text51.Text = pcent(26)  
form46.Text52.Text = pcent(27)  
form46.Text53.Text = pcent(28)  
form46.Text54.Text = pcent(29)  
form46.Text55.Text = pcent(30)  
form46.Text56.Text = pcent(31)  
form46.Text57.Text = x
```

```
form46.Text58.Text = pcsal(26)  
form46.Text59.Text = pcsal(27)  
form46.Text60.Text = pcsal(28)  
form46.Text61.Text = pcsal(29)  
form46.Text62.Text = pcsal(30)  
form46.Text63.Text = pcsal(31)
```



form46.Text64.Text = y

form46.Text9.Text = ocupent(26)  
form46.Text10.Text = ocupent(27)  
form46.Text11.Text = ocupent(28)  
form46.Text12.Text = ocupent(29)  
form46.Text13.Text = ocupent(30)  
form46.Text14.Text = ocupent(31)  
form46.Text15.Text = ocl

form46.Text16.Text = ocupsal(26)  
form46.Text17.Text = ocupsal(27)  
form46.Text18.Text = ocupsal(28)  
form46.Text19.Text = ocupsal(29)  
form46.Text20.Text = ocupsal(30)  
form46.Text21.Text = ocupsal(31)  
form46.Text22.Text = oc

form46.Text23.Text = intent(26)  
form46.Text24.Text = intent(27)  
form46.Text25.Text = intent(28)  
form46.Text26.Text = intent(29)  
form46.Text27.Text = intent(30)  
form46.Text28.Text = intent(31)  
form46.Text29.Text = inl

form46.Text30.Text = intsal(26)  
form46.Text31.Text = intsal(27)  
form46.Text32.Text = intsal(28)  
form46.Text33.Text = intsal(29)  
form46.Text34.Text = intsal(30)  
form46.Text35.Text = intsal(31)  
form46.Text36.Text = in

form46.Text37.Text = conent(26)  
form46.Text38.Text = conent(27)  
form46.Text39.Text = conent(28)  
form46.Text40.Text = conent(29)  
form46.Text41.Text = conent(30)  
form46.Text42.Text = conent(31)  
form46.Text43.Text = cnl

form46.Text44.Text = consal(26)  
form46.Text45.Text = consal(27)  
form46.Text46.Text = consal(28)  
form46.Text47.Text = consal(29)  
form46.Text48.Text = consal(30)  
form46.Text49.Text = consal(31)  
form46.Text50.Text = cn

form46.Text2.Text = 27  
form46.Text3.Text = 28

```
form46.Text4.Text = 29
form46.Text5.Text = 30
form46.Text6.Text = 31
form46.Text7.Text = 32
```

```
form46.Show
```

```
End Sub
```

```
Sub Command4_Click ()
Dim db As database
Dim dat As dynaset
Dim sal, ent As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
'Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop
```

```
Unload form10
Unload form44
```

```
form43.Text1.Text = 10
form43.Text2.Text = 11
form43.Text3.Text = 12
form43.Text4.Text = 13
form43.Text5.Text = 14
form43.Text6.Text = 15
form43.Text7.Text = 16
form43.Text8.Text = 17
form43.Text9.Text = 18
form43.Text10.Text = 19
form43.Text11.Text = 20
form43.Text12.Text = 21
form43.Text13.Text = 22
form43.Text14.Text = 23
form43.Text15.Text = 24
```

```
form43.Text16.Text = 25
form43.Text137.Text = 26
```

```
form43.Text18.Text = VEC(9)
form43.Text19.Text = VEC(10)
form43.Text20.Text = VEC(11)
form43.Text21.Text = VEC(12)
form43.Text22.Text = VEC(13)
form43.Text23.Text = VEC(14)
form43.Text24.Text = VEC(15)
form43.Text25.Text = VEC(16)
form43.Text26.Text = VEC(17)
```

```
form43.Text27.Text = VEC(18)
form43.Text28.Text = VEC(19)
form43.Text29.Text = VEC(20)
form43.Text30.Text = VEC(21)
form43.Text31.Text = VEC(22)
form43.Text32.Text = VEC(23)
form43.Text33.Text = VEC(24)
form43.Text17.Text = VEC(25)
```

```
form43.Text49.Text = abr(9)
form43.Text50.Text = abr(10)
form43.Text51.Text = abr(11)
form43.Text52.Text = abr(12)
form43.Text53.Text = abr(13)
form43.Text54.Text = abr(14)
form43.Text55.Text = abr(15)
form43.Text56.Text = abr(16)
form43.Text57.Text = abr(17)
```

```
form43.Text58.Text = abr(18)
form43.Text59.Text = abr(19)
form43.Text60.Text = abr(20)
form43.Text61.Text = abr(21)
form43.Text62.Text = abr(22)
form43.Text63.Text = abr(23)
form43.Text64.Text = abr(24)
form43.Text34.Text = abr(25)
```

```
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Set naset = db.CreateDynaset("Sis_Perdida")
Do While naset.EOF = False

    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            var(i) = naset("ndv")
            i = i + 1
        End If
    End If
    naset.MoveNext
Loop
```

```

Set naset = db.CreateDynaset("sis_perdida")
i = 0
ReDim bloq(35) As Variant
Do While naset.EOF = False
  If iniciov = naset("inicio") Then
    If abr(i) + "O" = naset("ruta") Then
      bloq(i) = naset("anblo")
      i = i + 1
    End If
  End If
  naset.MoveNext

```

Loop

```

Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
  If iniciov = naset("inicio") Then

    If abr(i) + "O" = naset("ruta") Then
      saliente(i) = naset("traff")
      i = i + 1
    End If

    If abr(j) + "I" = naset("ruta") Then
      entrante(j) = naset("traff")
      j = j + 1
    End If
  End If
  naset.MoveNext

```

Loop

```

Set naset = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While naset.EOF = False
  If iniciov = naset("inicio") Then
    If abr(i) + "O" = naset("ruta") Then
      cong(i) = naset("ccong")
      i = i + 1
    End If
  End If
  naset.MoveNext

```

Loop

halla la suma de traficos entrantes y salientes

```

ReDim total(35) As Variant

```

```

i = 0

```

```

For i = 0 To 35

```

total(i) = saliente(i) + entrante(i)  
Next i

form43.Text152.Text = total(9)  
form43.Text151.Text = total(10)  
form43.Text153.Text = total(11)  
form43.Text154.Text = total(12)  
form43.Text155.Text = total(13)  
form43.Text138.Text = total(14)  
form43.Text140.Text = total(15)  
form43.Text141.Text = total(16)  
form43.Text142.Text = total(17)

form43.Text143.Text = total(18)  
form43.Text144.Text = total(19)  
form43.Text145.Text = total(20)  
form43.Text146.Text = total(21)  
form43.Text147.Text = total(22)  
form43.Text114.Text = total(23)  
form43.Text116.Text = total(24)  
form43.Text113.Text = total(25)

form43.Text71.Text = var(9)  
form43.Text72.Text = var(10)  
form43.Text73.Text = var(11)  
form43.Text74.Text = var(12)  
form43.Text37.Text = var(13)  
form43.Text38.Text = var(14)  
form43.Text39.Text = var(15)  
form43.Text40.Text = var(16)  
form43.Text36.Text = var(17)

form43.Text41.Text = var(18)  
form43.Text42.Text = var(19)  
form43.Text43.Text = var(20)  
form43.Text44.Text = var(21)  
form43.Text45.Text = var(22)  
form43.Text46.Text = var(23)  
form43.Text47.Text = var(24)  
form43.Text35.Text = var(25)

form43.Text150.Text = cong(9)  
form43.Text156.Text = cong(10)  
form43.Text157.Text = cong(11)  
form43.Text158.Text = cong(12)  
form43.Text159.Text = cong(13)  
form43.Text160.Text = cong(14)  
form43.Text139.Text = cong(15)

form43.Text128.Text = cong(16)  
form43.Text129.Text = cong(17)

form43.Text130.Text = cong(18)  
form43.Text131.Text = cong(19)  
form43.Text132.Text = cong(20)  
form43.Text133.Text = cong(21)  
form43.Text134.Text = cong(22)  
form43.Text135.Text = cong(23)  
form43.Text136.Text = cong(24)  
form43.Text115.Text = cong(25)

form43.Text163.Text = saliente(9)  
form43.Text164.Text = saliente(10)  
form43.Text165.Text = saliente(11)  
form43.Text117.Text = saliente(12)  
form43.Text118.Text = saliente(13)  
form43.Text119.Text = saliente(14)  
form43.Text120.Text = saliente(15)  
form43.Text121.Text = saliente(16)  
form43.Text122.Text = saliente(17)  
form43.Text123.Text = saliente(18)  
form43.Text124.Text = saliente(19)

form43.Text125.Text = saliente(20)  
form43.Text126.Text = saliente(21)  
form43.Text127.Text = saliente(22)  
form43.Text99.Text = saliente(23)  
form43.Text98.Text = saliente(24)  
form43.Text106.Text = saliente(25)

form43.Text161.Text = bloq(9)  
form43.Text148.Text = bloq(10)  
form43.Text149.Text = bloq(11)  
form43.Text100.Text = bloq(12)  
form43.Text84.Text = bloq(13)  
form43.Text80.Text = bloq(14)  
form43.Text81.Text = bloq(15)  
form43.Text82.Text = bloq(16)  
form43.Text83.Text = bloq(17)

form43.Text91.Text = bloq(18)  
form43.Text108.Text = bloq(19)  
form43.Text109.Text = bloq(20)  
form43.Text107.Text = bloq(21)  
form43.Text110.Text = bloq(22)  
form43.Text111.Text = bloq(23)  
form43.Text112.Text = bloq(24)  
form43.Text70.Text = bloq(25)  
'para mostrar datos de traff

```
form43.Text162.Text = entrante(9)
form43.Text166.Text = entrante(10)
form43.Text167.Text = entrante(11)
form43.Text96.Text = entrante(12)
form43.Text95.Text = entrante(13)
form43.Text94.Text = entrante(14)
form43.Text93.Text = entrante(15)
form43.Text77.Text = entrante(16)
form43.Text76.Text = entrante(17)
```

```
form43.Text78.Text = entrante(18)
form43.Text79.Text = entrante(19)
form43.Text101.Text = entrante(20)
form43.Text102.Text = entrante(21)
form43.Text103.Text = entrante(22)
form43.Text104.Text = entrante(23)
form43.Text105.Text = entrante(24)
form43.Text97.Text = entrante(25)
'para mostrar ccong
Unload form44
form43.Show
```

**End Sub**

```
Sub Command1_Click ()
form1.Show
Unload form45
Unload FORM46
Unload form10

End Sub

Sub Command2_Click ()
form1.Show
Unload form45
Unload FORM46
Unload form10

End Sub

Sub Command3_Click ()
Unload FORM46
Dim db As database
Dim dat As dynaset
Dim sal, ert As Variant
Dim naset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dat = db.CreateDynaset("Datos")
Set naset = db.CreateDynaset("Sis_Perdida")
Dim i As Integer
Dim r As Integer
Dim var As Variant
ReDim VEC(35) As Variant
ReDim abr(35) As Variant
iniciov = form13.MaskedEdit1.Text
i = 0
Do While dat.EOF = False

    VEC(i) = dat("central")
    abr(i) = dat("abrev")

    i = i + 1
    dat.MoveNext
Loop

Unload form43
Unload form10

form44.Text5.Text = 27
form44.Text6.Text = 28
form44.Text7.Text = 29
form44.Text8.Text = 30
form44.Text9.Text = 31
form44.Text10.Text = 32
form44.Text18.Text = 32

form44.Text79.Text = VEC(26)
form44.Text80.Text = VEC(27)
```



```
form44.Text81.Text = VEC(28)
form44.Text1.Text = VEC(29)
form44.Text2.Text = VEC(30)
form44.Text3.Text = VEC(31)

form44.Text12.Text = abr(26)
form44.Text13.Text = abr(27)
form44.Text14.Text = abr(28)
form44.Text15.Text = abr(29)
form44.Text16.Text = abr(30)
form44.Text17.Text = abr(31)
ReDim var(35) As Variant
ReDim var1(35) As Variant
i = 0
Set naset = db.CreateDynaset("Sis_Perdida")
Do While naset.EOF = False

    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            var(i) = naset("ndv")
            i = i + 1
        End If
    End If
    naset.MoveNext
Loop

Dim sum As Variant
sum = 0
i = 0
For i = 0 To 35

    sum = sum + var(i)

Next i

Set naset = db.CreateDynaset("sis_perdida")
i = 0
ReDim bloq(35) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            bloq(i) = naset("anblo")
            i = i + 1
        End If
    End If
    naset.MoveNext

Loop

Dim sum1 As Variant
sum1 = 0
i = 0
For i = 0 To 35

    sum1 = sum1 + bloq(i)

Next i
```

```

Set naset = db.CreateDynaset("sis_perdida")
i = 0
j = 0
ReDim entrante(35) As Variant
ReDim saliente(35) As Variant
ReDim total(35) As Variant
Do While naset.EOF = False
    If iniciov = naset("inicio") Then

        If abr(i) + "O" = naset("ruta") Then
            saliente(i) = naset("traff")
            i = i + 1
        End If

        If abr(j) + "I" = naset("ruta") Then
            entrante(j) = naset("traff")
            j = j + 1
        End If
    End If
    naset.MoveNext
Loop

Dim sum2, sum3 As Variant
sum2 = 0
sum3 = 0
i = 0
For i = 0 To 35

    sum2 = sum2 + entrante(i)
    sum3 = sum3 + saliente(i)

Next i

Set naset = db.CreateDynaset("sis_perdida")
ReDim cong(35) As Variant
i = 0
Do While naset.EOF = False
    If iniciov = naset("inicio") Then
        If abr(i) + "O" = naset("ruta") Then
            cong(i) = naset("ccong")
            i = i + 1
        End If
    End If
    naset.MoveNext

Loop
Dim c As Variant

c = 0
i = 0
For i = 0 To 35

```

```
c = c + cong(i)
```

```
Next i
```

halla la suma de traficos entrantes y salientes

```
ReDim total(35) As Variant
```

```
Dim tot As Variant
```

```
i = 0
```

```
For i = 0 To 35
```

```
    total(i) = saliente(i) + entrante(i)
```

```
    tot = tot + total(i)
```

```
Next i
```

```
form44.Text54.Text = total(26)
```

```
form44.Text55.Text = total(27)
```

```
form44.Text56.Text = total(28)
```

```
form44.Text57.Text = total(29)
```

```
form44.Text58.Text = total(30)
```

```
form44.Text59.Text = total(31)
```

```
form44.Text60.Text = tot
```

```
form44.Text19.Text = var(26)
```

```
form44.Text20.Text = var(27)
```

```
form44.Text21.Text = var(28)
```

```
form44.Text22.Text = var(29)
```

```
form44.Text23.Text = var(30)
```

```
form44.Text24.Text = var(31)
```

```
form44.Text25.Text = sum
```

```
form44.Text72.Text = cong(26)
```

```
form44.Text73.Text = cong(27)
```

```
form44.Text74.Text = cong(28)
```

```
form44.Text75.Text = cong(29)
```

```
form44.Text76.Text = cong(30)
```

```
form44.Text77.Text = cong(31)
```

```
form44.Text78.Text = c
```

```
form44.Text47.Text = saliente(26)
```

```
form44.Text48.Text = saliente(27)
```

```
form44.Text49.Text = saliente(28)
```

```
form44.Text50.Text = saliente(29)
```

```
form44.Text51.Text = saliente(30)
```

```
form44.Text52.Text = saliente(31)
```

```
form44.Text53.Text = sum3
```

```
form44.Text33.Text = bloq(26)
```

---

```
form44.Text34.Text = bloq(27)
form44.Text35.Text = bloq(28)
form44.Text36.Text = bloq(29)
form44.Text37.Text = bloq(30)
form44.Text38.Text = bloq(31)
form44.Text39.Text = sum1
para mostrar datos de traff
```

```
form44.Text40.Text = entrante(26)
form44.Text41.Text = entrante(27)
form44.Text42.Text = entrante(28)
form44.Text43.Text = entrante(29)
form44.Text44.Text = entrante(30)
form44.Text45.Text = entrante(31)
form44.Text46.Text = sum2
para mostrar ccong
```

```
Unload form43
form44.Show
```

```
End Sub
```

```

Sub Command1_Click ()
form1.Show
Unload form9
form9.Hide
End Sub

```

```

Sub Command2_Click ()
form1.Show
Unload form9
form9.Hide

```

```

End Sub

```

```

Sub Command3_Click ()
Dim db As database
Dim dset As dynaset
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dset = db.CreateDynaset("carga_procesador_dia")
iniciov = form13.MaskedEdit1.Text
Do While dset.EOF = False
    If iniciov = dset("inicio") Then
        var = dset("horap")
        form27.Graph1.ThisPoint = Val(Mid$(dset("dia"), 1, 2))
        form27.Graph1.GraphData = dset("fchdi")
    End If
    dset.MoveNext
Loop
form9.Hide
form27.Show
*****PARA EL GRAFICO DE CIRCUITOS BIDIRECCIONALES EN HORA PICO

```

```

Dim dnaset As dynaset
Diru v As Variant
Set db = OpenDatabase("c:\vb\pruebas.mdb")
Set dnaset = db.CreateDynaset("trafico_saliente")
iniciov = form13.MaskedEdit1.Text
Do While dnaset.EOF = False
    If iniciov = dnaset("inicio") Then
        form27.Graph3.ThisPoint = Val(Mid$(dnaset("dia"), 1, 2))
        form27.Graph3.GraphData = dnaset("nblov")
    End If
    dnaset.MoveNext
Loop

```

```

End Sub

```

La idea de este anexo, es poder ofrecer al usuario técnico que tenga acceso a una central telefónica, una serie de comandos y valores específicos que pueden ser escritos en el prompt del computador que maneja la Central; y de esta forma obtener información sobre todas las mediciones descritas anteriormente.

Para el caso de la Central AXE 10, todos los resultados de las observaciones de tráfico se las guarda en archivos o subarchivos del IOG11 (dispositivo de entrada y salida de información), estos se encuentran definidos en el HD2 (disco duro de la central) en el volumen STATVOLUME. Los subarchivos pertenecen al archivo AFFILE.

Algunos archivos son:

TRARFILE, para mediciones de Tráfico por Rutas;  
TRATFILE, para mediciones de Tipo de Tráfico;  
TRDIPFILE, para mediciones de Dispersión de Tráfico;  
SEQSFILE, para mediciones de Calidad de Servicio.

El archivo AFFILE se divide en: AFFILE-00, AFFILE-01, AFFILE-02, AFFILE-03, y AFFILE-04.

Para extraer información de los archivos o subarchivos, se usa el comando:

```
IOFAT:FILE=file,NF;
```

Donde: FILE= archivo donde se direccionó el resultado;  
NF= se indica para que no imprima número de bloque.

A continuación tenemos, a manera de ejemplo, los diferentes comandos, los impresos, y las ejecuciones que se deben ingresar en el prompt del computador de la Central AXE 10, para obtener información de cada una de las seis mediciones; esto es:

#### MEDICIÓN DE TRAFICO POR RUTAS

Comandos:

TRRGI	Medición de tráfico en rutas, iniciación del grupo de registro.
TRRGE	Medición de tráfico en rutas, borrado del grupo de registro.
TRRGP	Medición de tráfico en rutas, impresión del grupo de registro.
TRRPI	Medición de tráfico en rutas, iniciación del programa de medición.
TRRPP	Medición de tráfico en rutas, impresión del programa de medición.
TRRIC	Medición de tráfico en rutas, cambio del direccionamiento de impresión.
TRIDP	Medición de tráfico en rutas, identidad del próximo programa de medición libre.
TRTSE	Registro de tráfico, finalización del programa de tiempo.
TRTSI	Registro de tráfico, iniciación del programa de tiempo.
TRTSP	Registro de tráfico, impresión del programa de tiempo.

**Impresos:**

TRAFFIC MEASUREMENT ON ROUTES GROUP DATA  
TRAFFIC MEASUREMENT ON ROUTES MEASURING PROGRAM  
TRAFFIC MEASUREMENT ON ROUTES RESULT, LSR  
TRAFFIC MEASUREMENT ON ROUTES RESULT, QSR  
TRAFFIC RECORDING TIME SCHEDULE  
TRAFFIC RECORDING IDENTITY

**Ejecución:**

El procedimiento es el mismo, tanto para Sistema de Pérdida o Sistema de Cola.

- 1.- Definir el grupo de registro

```
<TRRGI:TRG=8,R=STDO&STDI&CCLO&CCLI;
```

(R máximo puede ser hasta 32 rutas del mismo tipo)

- 2.- Imprimir el contenido de los grupos de registro

```
<TRRGP:TRG=trg;
```

- 3.- Solicitar un programa de medición libre

```
<TRDIP;
```

- 4.- Definir el programa de medición para la medición de tráfico por rutas

```
<TRRPI:MP=11,TRG=8;
```

- 5.- Imprimir los datos del programa de medición

```
<TRRPP:MP=ALL;
```

- 6.- Definir el programa de tiempo para el programa de medición

```
<TRTSI:MP=11, TIME=0930,RPL=60,NDAYS=365;
```

- 7.- Imprimir los valores del programa de tiempo

```
<TRTSP:MP=11;
```



8.- Interpretación de resultados.

MEDICION DEL TIPO DE TRAFICO

Comandos:

- TRIDP      Identidades de grabación de datos, imprimir.
- TRATI      Medición de tipo de tráfico, iniciación del programa de medida.
- TRATC      Medición de tipo de tráfico, cambio de direccionamiento de impresión.
- TRTSE      Tabla de tiempo de grabación de tráfico, finalización.
- TRTSI      Tabla de tiempo de grabación de tráfico, inicio.
- TRTSP      Tabla de tiempo de grabación de tráfico, imprimir.

Impresos:

- TRAFFIC RECORDING IDENTITY
- TRAFFIC RECORDING TIME SCHEDULE
- TRAFFIC TYPE MEASUREMENT PROGRAM
- TRAFFIC TYPE MEASUREMENT RESULT

Ejecución:

- 1.- Solicitar un programa de medida libre  
    <TRDIP;
- 2.- Definir el programa de medición para la medición de tipo de tráfico.  
    <TRATI:MP=12;
- 3.- Imprimir los datos de los programas de medición de tipo de tráfico.  
    <TRRPP:MP=ALL;

- 4.- Definir el programa de tiempo para el programa de medición  
<TRTSI:MP=12,TIME=0000,RPL=60,NDAYS=3,NRP=24,DATE=950702;
- 5.- Imprimir los valores del programa de tiempo  
<TRTSP:MP=12;
- 6.- Interpretación de los resultados.

#### MEDICION DE DISPERSION DE TRAFICO

##### Comandos:

TRIDP	Identidades de grabación de datos, imprimir.
TRDDI	Medición de dispersión de tráfico por destino, iniciación del programa de medida.
TRDGE	Medición de dispersión de tráfico, finalización del grupo de objetos de registro.
TRDGI	Medición de dispersión de tráfico, inicio del grupo de objetos de registro.
TRDIC	Medición de dispersión de tráfico, cambio del direccionamiento de impresión.
TRDME	Medición de dispersión de tráfico, finalización del programa de medida.
TRDMI	Medición de dispersión de tráfico, inicio del programa de medida.
TRDMP	Medición de dispersión de tráfico, impresión del programa de medida.
TRDRI	Medición de dispersión de tráfico por ruta, iniciación del programa de medida.
TRTSE	Tabla de tiempo de grabación de tráfico, finalizar.
TRTSI	Tabla de tiempo de grabación de tráfico, iniciar.

TRTSP      Tabla de tiempo de grabación de tráfico, imprimir.

Impresos:

TRAFFIC DISPERSION MEASUREMENT DESTINATION GROUP  
TRAFFIC DISPERSION MEASUREMENT ROUTE GROUP  
TRAFFIC DISPERSION MEASUREMENT PROGRAM DATA  
TRAFFIC DISPERSION MEASUREMENT PROGRAM SURVEY  
TRAFFIC DISPERSION MEASUREMENT PER ROUTE RESULT  
TRAFFIC DISPERSION MEASUREMENT PER DESTINATION RESULT  
TRAFFIC RECORDING IDENTITY  
TRAFFIC RECORDINGTIME SCHEDULE

Ejecución:

El procedimiento es igual para medición de dispersión por ruta o por destino, solo cambian los comandos TRDDI por TRDRI o viceversa.

Tráfico por destino

- 1.- Solicitar un programa de medida libre  
<TRIDP;
- 2.- Definir el programa de medición de dispersión de tráfico  
<TRDMI:MP=14,TRD;
- 3.- Definir los parámetros o criterios de selección  
<TRDDI:MP=14,TRD=1&4&6,R=STDO;
- 4.- Imprimir los datos del programa de medición de dispersión de tráfico  
<TRDMP:MP=14,DATA;
- 5.- Definir el programa de tiempo para el programa de medición

<TRTSI:MP=14,TIME=1600,RPL=30;

6.- Imprimir los valores del programa de tiempo

<TRTSP:MP=14;

7.- Interpretar los resultados

#### Tráfico por rutas

1.- Solicitar un programa de medida libre

<TRIDP;

2.- Definir el programa de medición de dispersión de tráfico por rutas

<TRDMI:MP=14,R2;

3.- Definir los parámetros o criterios de selección

<TRDRI:MP=14,R=STDO&QC40&QTS10,TRD=1;

4.- Imprimir los datos del programa de medición de dispersión de tráfico

<TRDMP:MP=14,DATA;

5.- Definir el programa de tiempo para el programa de medición

<TRTSI:MP=14,TIME=1600,RPL=30;

6.- Imprimir los valores para el programa de tiempo

<TRTSP:MP=14;

7.- Interpretar los resultados

#### MEDICION DE CARGA DEL PROCESADOR

Comandos:

PLSMI      Medición de carga del procesador, iniciación del programa de medida.

PLSMP      Medición de carga del procesador, impresión del programa de medida.

---

TRIDP      Identidades de grabación de datos, imprimir.  
TRTSE      Tabla de tiempo de grabación de tráfico, finalizar.  
TRTSI      Tabla de tiempo de grabación de tráfico, iniciar.  
TRTSP      Tabla de tiempo de grabación de tráfico, imprimir.

**Impresos:**

TRAFFIC RECORDING IDENTITY  
PROCESSOR LOAD MEASUREMENT PROGRAM  
TRAFFIC RECORDING TIME SCHEDULE  
PROCESSOR LOAD MEASUREMENT RESULT

**Ejecución:**

- 1.- Solicitar un programa de medida libre  
    <TRIDP;
- 2.- Definir el programa de medición de carga del procesador  
    <PLSMI:MP=13;
- 3.- Imprimir los datos de los programas para la medición de  
    carga del procesador  
    <PLSMP:MP=ALL;
- 4.- Definir el programa de tiempo para el programa de medición  
    <TRTSI:MP=13,TIME=0000,RPL=60,NDAYS=365,NRP=24,DATE=950702;
- 5.- Imprimir los valores del programa de tiempo  
    <TRTSP:MP=13;
- 6.- Interpretación de resultados

**ESTADISTICAS DE CALIDAD DE SERVICIO****Comandos:**

---

SQENE	Calidad de servicio, fin de exclusión de direcciones numéricas.
SQENI	Calidad de servicio, inicio de exclusión de direcciones numéricas.
SQENP	Calidad de servicio, impresión de series numéricas excluidas.
SQMPI	Calidad de servicio, inicio del programa de medida.
SQMPP	Calidad de servicio, impresión del programa de medida.
SQTLE	Calidad de servicio, fin de los límites de tiempo.
SQTLI	Calidad de servicio, inicio de los límites de tiempo.
SQTLP	Calidad de servicio, impresión de los límites de tiempo.
SQREP	Calidad de servicio, impresión de los resultados.
TRIDP	Identidades de grabación de datos, imprimir.
TRTSE	Tabla de tiempo de grabación de tráfico, finalizar.
TRTSI	Tabla de tiempo de grabación de tráfico, iniciar.
TRTSP	Tabla de tiempo de grabación de tráfico, imprimir.

**Impresos:**

SERVICE QUALITY STATISTICS MEASURING PROGRAM  
SERVICE QUALITY STATISTICS RESULT  
SERVICE QUALITY STATISTICS TIME LIMITS  
SERVICE QUALITY STATISTICS EXCLUSION OF NRD  
TRAFFIC RECORDING TIME SCHEDULE  
TRAFFIC RECORDING IDENTITY

**Ejecución:**

1.- Solicitar un programa de medida libre

- <TRIDP;
- 2.- Definir los límites de tiempo  
<SQTLI:ANSWER=5&10;  
<SQTLI:CTIME=30&120;
  - 3.- Imprimir los límites de tiempo  
<SQTLP;
  - 4.- Hacer la exclusión de las series que no deben entrar en la medición  
<SQENI:NRL=2,NRD=57&61&25;
  - 5.- Imprimir las series numéricas excluidas  
<SQENP;
  - 6.- Definir el programa de medición para estadísticas de calidad de servicio  
<SQMPI:MP=11,QTA=1000,R2=MS50;
  - 7.- Imprimir los datos de los programas de medición de las estadísticas de calidad de servicio  
<SQMPP:MP=ALL;
  - 8.- Definir el programa de tiempo para el programa de medición  
<TRTSI:MP=11,TIME=0930,RPL=120,DATE=950702;
  - 9.- Imprimir los valores del programa de tiempo  
<TRTSP:MP=11;
  10. Interpretación de los resultados

#### MEDICION DE OBSERVACION DE TRAFICO

Comandos:

- TOBSI      Observación de tráfico, inicio.  
TOBSP      Observación de tráfico, criterios de selección,

imprimir.

TOCOR Observación de tráfico, criterios de selección,  
finalizar.

TOCOS Observación de tráfico, criterios de selección,  
iniciar.

Impresos:

TRAFFIC OBSERVATION RESULT

TRAFFIC OBSERVATION SELECTION CRITERIA

Ejecución:

1.- Definir los criterios de selección para la observación de tráfico

<TOCOS:QTA=10,R2=STDO;

2.- Imprimir los criterios de selección para la observación de tráfico

<TOBSP;

3.- Iniciar la observación de tráfico

<TOBSI:NOFILE;

MONITORING EQUIPMENT CONNECTED?

:Y;

.

.

.

PARAMETERS

:1;

CONTINUATION?



:N)

Si cuando aparece la pregunta "CONTINUATION?", se escribe "Y", la observación continuará hasta completar la cuota llamada QTA; caso contrario, si se escribe "N", la medición quedará suspendida hasta volver a mandar el comando TOBSI.

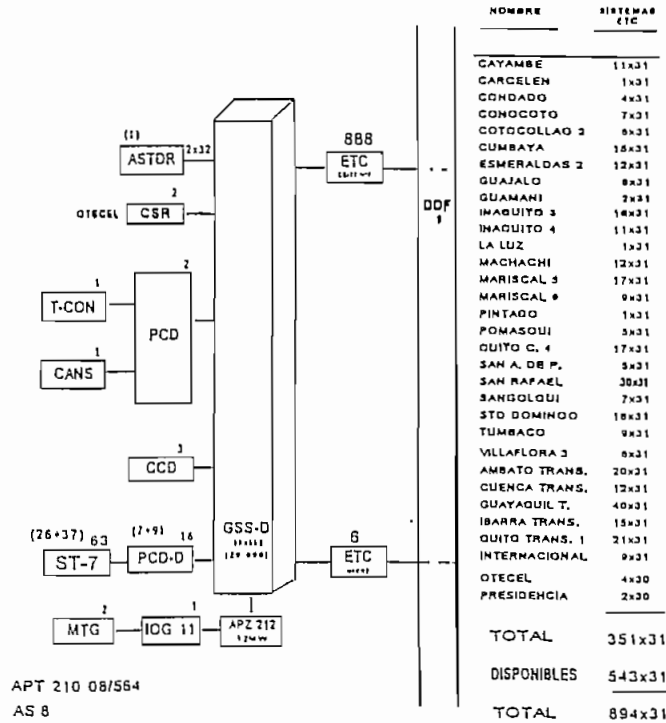
4.- Interpretación de los resultados.

NOTA.- Las sentencias escritas en el prompt del computador, tienen una secuencia definida en programa propio de la Central AXE 10 de Ericsson, que maneja el Lenguaje Basic del mencionado fabricante.

INFORMACIONES ADICIONALES.- Es necesario dar a conocer algunos aspectos complementarios, a fin de que el usuario técnico de la Central Telefónica, pueda tener una idea global de todo el sistema AXE 10 respecto a otras centrales. Así, tenemos:

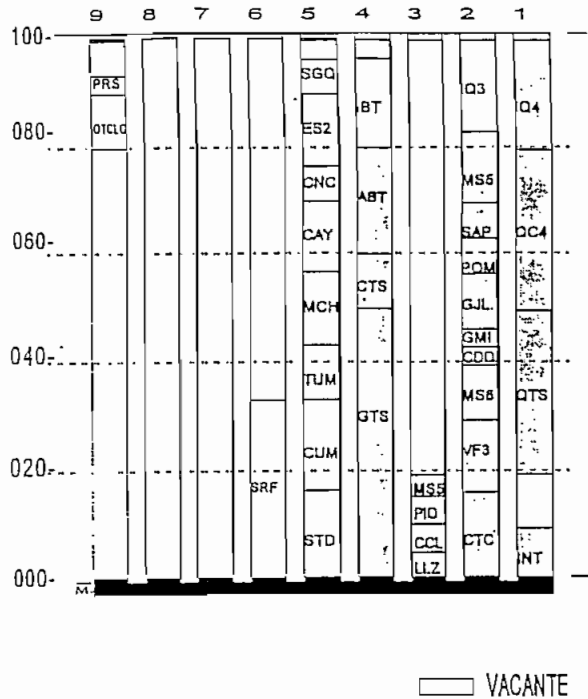
- \* Diagrama de Enlaces
- \* Distribución del DDF
- \* Configuración de los IOs.

## DIAGRAMA DE ENLACES.



En este diagrama se especifican las conexiones de la central desde el punto de vista interno como externo ademas se da informacion sobre la version del sistema de aplicacion y de APT.

## CONFIGURACION DEL DDF.



En este diagrama se muestra la distribucion del DDF indicando las posiciones que tiene libres y detallando las ocupadas.